



TUGAS AKHIR - KI091391

Rancang Bangun Visualisasi Peta Kerjasama Peneliti Berdasarkan Topik Penelitian pada Sistem Repositori Peneliti menggunakan Pemodelan Graf

AGUS NUGROHO
NRP 5110 100 124

Dosen Pembimbing I
Umi Laili Yuhana, S.Kom., M.Sc.

Dosen Pembimbing II
Diana Purwitasari, S.Kom., M.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2014



FINAL PROJECT - KI091391

DESIGN AND IMPLEMENTATION OF VISUALIZING CORRELATED RESEARCHER MAP BASED ON GRAPH MODEL OF RESEARCH TOPICS

AGUS NUGROHO
NRP 5110 100 124

Advisor I
Umi Laili Yuhana, S.Kom., M.Sc.

Advisor II
Diana Purwitasari, S.Kom., M.Sc.

INFORMATICS ENGINEERING DEPARTEMENT
Information Technology Faculty
Sepuluh Nopember Institute of Technology
Surabaya 2014

LEMBAR PENGESAHAN

**Rancang Bangun Visualisasi Peta Kerjasama Peneliti
Berdasarkan Topik Penelitian pada Sistem Repositori
Peneliti menggunakan Pemodelan Graf**

TUGAS AKHIR

**Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember**

**Oleh :
AGUS NUGROHO
NRP : 5110 100 124**

Disetujui oleh Dosen Pembimbing Tugas Akhir :

**Umi Laili Yuhana, S.Kom., M.Sc.
NIP: 197906262005012002**

(pembimbing 1)

**Diana Purwitasari, S.Kom., M.Sc.
NIP: 197804102003122001**

(pembimbing 2)

**SURABAYA
JULI 2014**

Rancang Bangun Visualisasi Peta Kerjasama Peneliti Berdasarkan Topik Penelitian pada Sistem Repositori Peneliti menggunakan Pemodelan Graf

Nama Mahasiswa : Agus Nugroho
NRP : 5110 100 124
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing I : Umi Laili Yuhana, S.Kom, M.sc.
Dosen Pembimbing II : Diana Purwitasari, S.Kom, M.sc.

ABSTRAK

Bila suatu negara ingin bersaing di dunia global yang persaingannya semakin ketat sekarang ini, maka penelitian dan pengembangan harus menjadi perhatian utama. Peluang penelitian dimasa depan akan menjadi peluang bagi kemajuan suatu bangsa. Informasi mengenai peta kerjasama peneliti menjadi penting bagi para peneliti untuk memprediksi peluang kerjasama dengan peneliti lain.

Sistem Repositori Peneliti Institut Teknologi Sepuluh Nopember (ITS) merupakan sistem informasi yang secara khusus memberikan informasi kepada para peneliti, mahasiswa maupun masyarakat umum seputar dunia penelitian di ITS. Aplikasi ini memiliki salah satu fitur yang memberikan informasi kepada pengguna mengenai peta kerjasama peneliti di ITS. Pengguna dapat melihat kedekatan kerjasama peneliti berdasarkan topik penelitiannya. Peta kerjasama peneliti ini divisualisasikan menggunakan pemodelan graf, sehingga pengguna akan mudah melihat kedekatan-kedekatan para peneliti. Setiap peneliti direpresentasikan dengan lingkaran (node) pada graf, sedangkan hubungan kedekatan antar peneliti direpresentasikan dengan garis (edge). Semakin besar node, berarti semakin banyak penelitian yang telah dilakukan oleh peneliti tersebut, dan sebaliknya. Selain itu, warna node menunjukkan area penelitian peneliti. Semakin tebal edge, berarti semakin dekat hubungan kerjasama antar node (peneliti).

Fitur visualisasi peta kerjasama peneliti berdasarkan topik penelitian ini dapat menyediakan informasi peta kerjasama peneliti di ITS yang dibutuhkan pengguna. Informasi peta kerjasama peneliti yang

ditampilkan dapat membantu pengguna untuk mengetahui perkembangan penelitian yang ada di ITS.

Kata kunci: *Institut Teknologi Sepuluh Nopember, Graf, Peta Kerjasama Peneliti, Topik Penelitian, Visualisasi*

Design and Implementation of Visualizing Correlated Researcher Map Based on Graph Model of Research Topics

Student Name : Agus Nugroho
NRP : 5110 100 124
Major : Informatics Engineering FTIf-ITS
Advisor I : Umi Laili Yuhana, S.Kom, M.sc.
Advisor II : Diana Purwitasari, S.Kom, M.sc.

ABSTRACT

If the a country wants to compete in the global world with increasingly fierce competition now, the research and development should be a major concern. The opportunities of research in the future will be an opportunity for the advancement of a nation. Information on the map of cooperation of researchers to be important for researchers to predict the opportunities for cooperation with other researchers.

Sistem Repositori Peneliti Institut Teknologi Sepuluh Nopember (ITS) is an information system that specifically provide information to researchers, students and public society about the research in ITS. This application provide a feature that provides information to users about the map of cooperation of researchers in ITS. Users can see the closeness of cooperation of researchers based on their research topic. Map of this research cooperation is visualized using a graph modeling, so users will be easy to see closeness the researchers. Each researcher is represented by a circle (node) in the graph, while the closeness of the relationship between researchers represented by a line (edge). The larger the node, meaning more and more research has been done by researchers, and vice versa. In addition, the node color indicates of research area. The thicker edge, meaning the closer cooperation between nodes (the researcher).

Map visualization features based on the topic of cooperation of researchers of this study can provide map information of research cooperation at ITS that required by users. The information map of researcher cooperation displayed can help the user to know the progress of research in the ITS.

Keyword: *Institut Teknologi Sepuluh Nopember, Graph, Map of Cooperation Researcher, Topic of Research, Visualization*

KATA PENGANTAR

Sesungguhnya segala puji hanya milik Allah, kita memuji-Nya, kita meminta pertolongan-Nya, kita meminta ampunan-Nya serta kita berlindung kepada Allah dari kejahatan jiwa kita dan dari keburukan amalan kita. Siapa orangnya yang Allah berikan hidayah kepadanya, maka tiada seorang-pun yang bisa menyesatkannya, dan siapa orangnya yang Allah sesatkan, maka tiada seorang-pun yang bisa memberikan hidayah kepadanya.

Puji syukur kepada Allah Tabaraka wa Ta'ala atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

Rancang Bangun Visualisasi Peta Kerjasama Peneliti Berdasarkan Topik Penelitian pada Sistem Repositori Peneliti menggunakan Pemodelan Graf

Melalui lembar ini, penulis hanya ingin menyampaikan ucapan terima kasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah Tabaraka wa Ta'ala atas segala nikmat dan rahmat yang telah diberikan selama ini.
2. Kedua orang tua penulis yang tiada henti-hentinya mencurahkan kasih sayang, perhatian, dan doa kepada penulis selama ini.
3. Ibu Yuhana dan Ibu Diana selaku dosen pembimbing yang telah memberikan bimbingan, motivasi, dan meluangkan waktu untuk membantu pengerjaan Tugas Akhir ini dengan penuh kesabaran.

4. Ibu Bilqis selaku dosen wali yang telah memberikan arahan, nasehat, dan motivasi kepada penulis selama menempuh studi di Teknik Informatika ITS.
5. Direktorat Jenderal Pendidikan Tinggi, Kementerian Pendidikan dan Kebudayaan yang telah memberikan dukungan finansial melalui Beasiswa Bidik Misi tahun 2010-2014.
6. Sahabat dekat yang selalu memberikan motivasi dan inspirasi untuk terus berkarya: Bang Irfan, Bang Gandha, Bang Kresna, Bang Fahad, Bang Agro, Bang Yudha, Bang Andi, Bang Fandi, Mas Sholeh, Taufiq, Ahmad, Imam, Sulaiman, Faiz, Fahmi, dan Azis.
7. Teman dekat yang selalu memberikan dukungan dan berbagi ilmu kepada penulis selama menempuh studi di Teknik Informatika ITS: Abrian, Fadli, Yasin, Riduwan, Fadjar, Muamar, Amin, Hani, Fahmi, Vina, Luluk, Anita, Dhila, dan Dika.
8. Keluarga besar Laboratorium Sistem Cerdas, keluarga besar Laboratorium Rekayasa Perangkat Lunak, dan keluarga besar Keluarga Muslim Informatika.
9. Serta pihak-pihak lain yang namanya tidak dapat penulis sebutkan satu -persatu.

Bagaimanapun juga penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2014

Agus Nugroho

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK.....	ix
ABSTRACT	xi
KATA PENGANTAR	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR.....	xxi
DAFTAR TABEL.....	xxv
DAFTAR KODE SUMBER	xxvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Metodologi	4
1.7 Sistematika	6
BAB II TINJAUAN PUSTAKA	9
2.1 Sistem Repositori Peneliti Institut Teknologi Sepuluh Nopember (ITS) Surabaya.....	9
2.2 Web Personal Dosen Institut Teknologi Sepuluh Nopember (ITS) Surabaya.....	11
2.3 CodeIgniter	12
2.4 Zend Search Lucene Framework Library	13

2.5	Vis.Js.....	15
2.6	JavaScript.....	15
2.7	JavaScript Object Notation	16
2.8	Algoritma TF-IDF (Term Frequency - Inversed Document Frequency).....	17
2.9	Algoritma Vector Space Model	20
2.10	Algoritma Dice's coefficient	23
2.11	Referensi aplikasi yang sudah pernah ada	23
2.11.1	Mapping Breivik's mind.....	23
2.11.2	Immersion.....	24
BAB III ANALISIS DAN PERANCANGAN		27
3.1.	Analisis	27
3.1.1.	Permasalahan.....	27
3.1.2.	Deskripsi Umum Perangkat Lunak.....	29
3.1.3.	Arsitektur Sistem	31
3.1.3	Kebutuhan Fungsional.....	34
3.2	Perancangan	46
3.2.1	Perancangan Basis Data	46
3.2.2	Perancangan Antarmuka.....	53
3.2.3	Perancangan Proses Aplikasi.....	58
BAB IV IMPLEMENTASI.....		67
4.1	Lingkungan Implementasi	67
4.1.1	Lingkungan Implementasi Perangkat Keras.....	67
4.1.2	Lingkungan Implementasi Perangkat Lunak.....	67

4.2	Implementasi Antarmuka	67
4.2.1	Antarmuka Halaman Visualisasi Peta Kerjasama Penelit.....	68
4.2.2	Antarmuka Halaman Login Manajemen Sistem Repositori Peneliti	70
4.2.3	Antarmuka Halaman Manajemen Visualisasi Sistem Repositori Peneliti	72
4.3	Implementasi Konsep MVC	74
4.3.1	Implementasi Model.....	74
4.3.2	Implementasi View.....	75
4.3.3	Implementasi Controller.....	76
4.4	Implementasi Basis Data	77
4.4.1.	Implementasi Struktur Basis Data.....	77
4.4.2.	Implementasi <i>Query</i>	82
4.5	Implementasi Proses Aplikasi	101
4.5.1	Proses Pengindeksan Data Penelitian Tugas Akhir.	101
4.5.2	Proses Tokenisasi dan Menghilangkan Stopwords .	102
4.5.3	Proses Stemming	103
4.5.4	Proses Pembobotan Kata (Term).....	104
4.5.5	Proses Penghitungan Bobot Topik Antar Peneliti ...	105
4.5.6	Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti.....	105
4.5.7	Proses Penghitungan Bobot Kerjasama Berdasarkan Topik Penelitian Antar Peneliti	106
4.5.8	Proses Pemodelan Graf Peta Kerjasama Peneliti Dalam Bentuk JSON	107

4.5.9 Proses Visualisasi Graf Berbasis Web.....	109
4.5.10 Proses Login Manajemen Sistem Repositori Peneliti	
117	

BAB V PENGUJIAN DAN EVALUASI119

5.1. Lingkungan Pelaksanaan Pengujian	119
5.2. Dasar Pengujian.....	119
5.3. Pengujian Fungsionalitas	119
5.3.1 Pengujian Melihat Peta Kerjasama Peneliti Semua Fakultas.....	120
5.3.2 Pengujian Melihat Peta Kerjasama Peneliti Setiap Fakultas.....	121
5.3.3 Pengujian Melihat Persentase Persebaran Peneliti Setiap Fakultas.....	123
5.3.4 Pengujian Melakukan Pencarian Peneliti	124
5.3.5 Pengujian Melihat Detail Peneliti.....	128
5.3.6 Pengujian Memperbarui Data Peta Kerjasama Peneliti	
129	
5.4 Pengujian Kegunaan	133
5.4.1 Kriteria Responden.....	133
5.4.2 Lingkungan Pengujian Kegunaan.....	133
5.4.3 Skenario Pengujian Kegunaan.....	133
5.4.4 Daftar Responden	134
5.4.5 Hasil Pengujian Kegunaan	135
5.4 Evaluasi Pengujian.....	137
5.4.1 Evaluasi Pengujian Fungsionalitas	137

5.4.2 Evaluasi Pengujian Kegunaan	138
BAB VI PENUTUP	141
6.1. Kesimpulan.....	141
6.2. Saran.....	141
DAFTAR PUSTAKA	143
LAMPIRAN A – FORMULIR PENENTUAN <i>THRESHOLD</i> PETA KERJASAMA PENELITI.....	145
LAMPIRAN B – FORMULIR PENILAIAN PENGUJIAN KEGUNAAN.....	149
LAMPIRAN C – PERANCANGAN BASIS DATA CONCEPTUAL DATA MODEL (CDM) DAN PHYSICAL DATA MODEL (PDM).....	151
LAMPIRAN D – HASIL SURVEI PENENTUAN <i>THRESHOLD</i> PETA KERJASAMA PENELITI.....	153
LAMPIRAN E – DIAGRAM ALIR KAKAS VISUALISASI	155
BIODATA PENULIS.....	157

DAFTAR TABEL

Tabel 2. 1 Contoh Perhitungan TF-IDF	19
Tabel 2. 2 Perhitungan vector space model.....	21
Tabel 2. 3 Tabel hasil nilai cosinus setiap dokumen	22
Tabel 3. 1 Deskripsi Kasus Penggunaan Visualisasi Peta Kerjasama Peneliti.....	35
Tabel 3. 2 Rincian Alur Kasus Penggunaan UC-P01	36
Tabel 3. 3 Rincian Alur Kasus Penggunaan UC-P02.....	38
Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-P03	40
Tabel 3. 5 Rincian Alur Kasus Penggunaan UC-P04.....	40
Tabel 3. 6 Rincian Alur Kasus Penggunaan UC-P05	43
Tabel 3. 7 Rincian Alur Kasus Penggunaan UC-P06.....	44
Tabel 3. 8 Atribut Tabel tran_bobot_katadasar	46
Tabel 3. 9 Atribut Tabel tran_bobot_kata_indeks	47
Tabel 3. 10 Atribut Tabel tran_bobot_kata_dokumen	48
Tabel 3. 11 Atribut Tabel tran_bobot_vektor_topik.....	49
Tabel 3. 12 Atribut Tabel tran_bobot_vektor_1	49
Tabel 3. 13 Atribut Tabel tran_bobot_vektor_2.....	50
Tabel 3. 14 Atribut Tabel tran_bobot_dosen_single	51
Tabel 3. 15 Atribut Tabel tran_bobot_dosen_peneliti.....	52
Tabel 3. 16 Atribut Tabel tref_user_admin	52
Tabel 5. 1 Melihat Peta Kerjasama Peneliti Semua Fakultas	120
Tabel 5. 2 Melihat Peta Kerjasama Peneliti Setiap Fakultas	122
Tabel 5. 3 Melihat Persebaran Peneliti Setiap Fakultas	123
Tabel 5. 4 Melakukan Pencarian Peneliti	125
Tabel 5. 5 Melihat Detail Peneliti	128
Tabel 5. 6 Memperbarui Data Peta Kerjasama Peneliti	129
Tabel 5. 7 Responden Pengujian Kegunaan Visualisasi Peta Kerjasama Peneliti.....	134
Tabel 5. 8 Rekapitulasi Penilaian Antarmuka Pengguna	135

Tabel 5. 9 Rekapitulasi Penilaian Informasi Graf Peta Kerjasama
Peneliti.....136

Tabel 5. 10 Rekapitulasi Penilaian Informasi Ketepatan Lokasi
.....137

Tabel 5. 11 Rekapitulasi Akhir Pengujian Kegunaan.....139

Tabel D. 1 Hasil Survei Penentuan Threshold Bobot Antar Peneliti
.....153

Tabel D. 2 Hasil Survei Penentuan Jumlah Mode Pada Graf
Visualisasi.....153

DAFTAR KODE SUMBER

Kode Sumber 4. 1 Kode Sumber Controller Halaman Visualisasi Peta Kerjasama Peneliti.....	70
Kode Sumber 4. 2 Kode Sumber Controller Halaman Login Manajemen Sistem Repositori Peneliti	72
Kode Sumber 4. 3 Kode Sumber Halaman Manajemen Visualisasi Sistem Repositori Peneliti	74
Kode Sumber 4. 4 Kode Sumber Kelas Model Index_model	75
Kode Sumber 4. 5 Implementasi View Tema Header Halaman Visualisasi Peta Kerjasama Peneliti	76
Kode Sumber 4. 6 Implementasi Kelas Controller Manage.....	77
Kode Sumber 4. 7 Implementasi Struktur Basis Data Tabel tran_bobot_katadasar.....	78
Kode Sumber 4. 8 Implementasi Struktur Basis Data Tabel tran_bobot_kata_indeks.....	79
Kode Sumber 4. 9 Implementasi Struktur Basis Data Tabel tran_bobot_kata_dokumen	79
Kode Sumber 4. 10 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_topik	80
Kode Sumber 4. 11 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_1	80
Kode Sumber 4. 12 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_2	81
Kode Sumber 4. 13 Implementasi Struktur Basis Data Tabel tran_bobot_dosen_single.....	81
Kode Sumber 4. 14 Implementasi Struktur Basis Data Tabel tran_bobot_dosen_peneliti	82
Kode Sumber 4. 15 Implementasi Struktur Basis Data Tabel tref_user_admin.....	82
Kode Sumber 4. 16 Implementasi Query Mengosongkan Semua Tabel Penghitungan Peta Kerjasama Peneliti.....	83

Kode Sumber 4. 17 Implementasi Query Mendapatkan Jumlah Dokumen Tugas Akhir	84
Kode Sumber 4. 18 Implementasi Query Mendapatkan Data Tugas Akhir.....	84
Kode Sumber 4. 19 Implementasi Query Mendapatkan Kata Dasar	85
Kode Sumber 4. 20 Implementasi Query Menyimpan Daftar Term Unik	86
Kode Sumber 4. 21 Implementasi Query Mendapatkan Daftar Term Unik Berdasarkan Frekuensi Kemunculan	87
Kode Sumber 4. 22 Implementasi Query Menyimpan Daftar Term Beserta Frekuensi Munculnya Untuk Setiap Dokumen.....	88
Kode Sumber 4. 23 Implementasi Query Memperbarui TF-IDF Setiap Term	89
Kode Sumber 4. 24 Implementasi Query Menyimpan Term Berdasarkan ID Peneliti.....	90
Kode Sumber 4. 25 Implementasi Query Menyimpan Daftar Term Teratas Setiap Peneliti	91
Kode Sumber 4. 26 Implementasi Query Mendapatkan Daftar Term Semua Peneliti	91
Kode Sumber 4. 27 Implementasi Query Menyimpan Hubungan Antar Peneliti Berdasarkan ID Peneliti	92
Kode Sumber 4. 28 Implementasi Query Mendapatkan Hubungan Antar Peneliti Berdasarkan ID Peneliti	93
Kode Sumber 4. 29 Implementasi Query Memperbarui Hubungan Bobot Topik Antar Peneliti	94
Kode Sumber 4. 30 Implementasi Query Memperbarui Hubungan Bobot Kerjasama Antar Peneliti	94
Kode Sumber 4. 31 Implementasi Query Memperbarui Bobot Final Hubungan Antar Peneliti	95
Kode Sumber 4. 32 Implementasi Query Mendapatkan Data Setiap Peneliti.....	96

Kode Sumber 4. 33 Implementasi Query Mendapatkan Fakultas Peneliti.....	97
Kode Sumber 4. 34 Implementasi Query Mendapatkan Jurusan Peneliti.....	97
Kode Sumber 4. 35 Implementasi Query Memperbarui ID Node Peneliti.....	98
Kode Sumber 4. 36 Implementasi Query Memperbarui Jumlah Peneliti yang Mirip untuk Setiap Peneliti.....	99
Kode Sumber 4. 37 Implementasi Query Mendapatkan Bobot Hubungan Kerjasama Berdasarkan Kemiripan Topik Penelitian Antar Peneliti.....	100
Kode Sumber 4. 38 Implementasi Query Mendapatkan Bobot Hubungan Antar Peneliti Paling Tinggi	100
Kode Sumber 4. 39 Implementasi Query Mendapatkan Node Peneliti.....	101
Kode Sumber 4. 40 Kode Sumber Proses Pengindeksan Data Penelitian Tugas Akhir.....	102
Kode Sumber 4. 41 Kode Sumber Proses Menghilangkan Stopwords.....	103
Kode Sumber 4. 42 Kode Sumber Proses Melakukan Stemming	104
Kode Sumber 4. 43 Kode Sumber Proses Pembobotan Kata....	104
Kode Sumber 4. 44 Kode Sumber Proses Penghitungan Bobot Relasi Topik Antar Peneliti	105
Kode Sumber 4. 45 Kode Sumber Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti.....	106
Kode Sumber 4. 46 Kode Sumber Proses Penghitungan Bobot Kerjasama Berdasarkan Topik Penelitian Antar Peneliti	107
Kode Sumber 4. 47 Kode Sumber Proses Pemodelan Graf Peta Kerjasama Peneliti Dalam Bentuk JSON.....	109
Kode Sumber 4. 48 Kode Sumber Proses Inisiasi Visualisasi Graf Peta Kerjasama Peneliti.....	111

Kode Sumber 4. 49 Kode Sumber Proses Menampilkan Detail
Peneliti dan Mengurutkan Peneliti-Peneliti yang Mirip Topik
Penelitiannya113

Kode Sumber 4. 50 Kode Sumber Proses Menghitung dan
Menampilkan Persentase Persebaran Peneliti115

Kode Sumber 4. 51 Kode Sumber Proses Pencarian Peneliti....116

Kode Sumber 4. 52 Proses Menampilkan Peta Kerjasama Peneliti
Setiap Area Peneliti117

Kode Sumber 4. 53 Kode Sumber Proses Login Manajemen Sistem
Repositori Peneliti118

BAB I

PENDAHULUAN

Bagian ini akan dijelaskan hal-hal yang menjadi latar belakang, permasalahan yang dihadapi, batasan masalah, tujuan dan manfaat, metodologi, dan sistematika penulisan yang digunakan dalam pembuatan Tugas Akhir.

1.1 Latar Belakang

Pada era pengetahuan sekarang ini, kemajuan industri suatu negara tidak hanya dipengaruhi oleh sumber daya alam melimpah yang dimilikinya, tetapi justru lebih ditentukan oleh sumber daya manusia serta ilmu dan teknologi yang dikuasainya. Bila suatu negara ingin bersaing di dunia global yang persaingannya semakin ketat sekarang ini, maka penelitian dan pengembangan harus menjadi perhatian utama [1]. Selain itu, peluang penelitian di masa depan akan menjadi celah bagi kemajuan suatu bangsa. Salah satunya, peluang ini bisa didapatkan dengan melakukan identifikasi bidang utama dari disiplin ilmu dan publikasi penelitian yang ada saat ini [2]. Dari identifikasi tersebut nantinya dapat dipetakan adanya hubungan dari beberapa elemen penelitian seperti para peneliti, publikasi ilmiah serta topik penelitian. Kemudian dari pemetaan tersebut akan terlihat peluang-peluang yang ada misalnya saja peluang kerjasama antar peneliti dimasa depan, peluang topik penelitian baru dan lain sebagainya. Namun, di Indonesia saat ini masih belum banyak lembaga yang menyajikan informasi mengenai pemetaan kerjasama peneliti.

Perguruan Tinggi merupakan salah satu lembaga penelitian yang paling berpengaruh terhadap kemajuan dunia penelitian disuatu negara. Hal ini karena banyaknya penyumbang publikasi penelitian berasal dari Perguruan Tinggi. Saat ini untuk menggiatkan dunia penelitian di Indonesia, Institut Teknologi Sepuluh Nopember (ITS) Surabaya sedang mengembangkan sebuah sistem informasi yang akan menunjang dunia penelitian

khususnya di lingkungan ITS. Sistem informasi tersebut yaitu Sistem Repositori Peneliti. Sistem Repositori Peneliti atau disebut juga dengan Resources ITS (RESITS), nantinya akan menjadi versi terbaru dari Sistem Informasi Personal Dosen yang sebelumnya sudah ada [3]. Pada Web Personal Dosen belum tersedia fitur yang secara khusus memberikan informasi tentang kerjasama antar peneliti di ITS. Sistem Repositori Peneliti secara khusus akan memberikan informasi kepada masyarakat seputar dunia penelitian yang ada di ITS. Beberapa fitur yang terdapat dalam sistem informasi tersebut yaitu pengguna dapat melakukan pencarian peneliti dengan kriteria tertentu, melihat daftar publikasi jurnal penelitian terakhir, melihat hubungan kerjasama antar peneliti, serta fitur lainnya.

Pada saat ini, bentuk representasi visual kerjasama (jaringan) yang dinilai paling modern adalah menggunakan visualisasi berbentuk graf [2]. Dengan pemodelan graf, informasi kerjasama akan lebih mudah dipahami dibandingkan dengan bentuk visualisasi lainnya seperti grafik, tabel, dan *listview*. Dalam sebuah visualisasi berbentuk graf, *node* biasanya diwakili oleh bentuk lingkaran dengan ukuran dan warna berbeda-beda yang menunjukkan karakteristik masing-masing *node*. Dan setiap *edge* yang diwakili oleh garis biasanya memiliki panjang dan ketebalan yang berbeda-beda pula, sesuai karakteristik hubungan antar *node* yang dihubungkan oleh *edge* tersebut.

Untuk itu, muncul sebuah gagasan untuk mengembangkan sebuah perangkat lunak yang dapat digunakan para peneliti, mahasiswa, maupun masyarakat umum untuk mengetahui peta kerjasama peneliti yang ada di ITS. Agar mudah dipahami, peta kerjasama antar peneliti divisualisasikan menggunakan pemodelan graf. Perangkat lunak yang akan dibuat ini akan menjadi bagian dari fitur Sistem Repositori Peneliti. Perangkat lunak ini hanya akan berfokus pada peta kerjasama antar peneliti berdasarkan topik penelitian yang diolah dari dokumen tugas akhir. Topik penelitian setiap peneliti diekstraksi dari publikasi penelitian yang pernah

dibuat oleh peneliti. Kemudian hubungan antar peneliti didapatkan berdasarkan kemiripan topik penelitian serta kerjasama yang pernah dilakukan bersama.

Harapan yang ingin dicapai dalam tugas akhir ini, para peneliti, mahasiswa ataupun masyarakat umum dapat memperoleh informasi peta kerjasama antar peneliti berdasarkan kemiripan topik penelitian masing-masing peneliti yang ada di ITS dengan visualisasi yang informatif serta mudah dipahami.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut :

1. Bagaimana mengetahui peneliti yang memiliki kesamaan berdasarkan topik dari penelitian yang pernah dibuat?
2. Bagaimana memodelkan peneliti-peneliti yang mirip berdasarkan topik menggunakan visualisasi pemodelan graf berbasis *web*?

1.3 Batasan Masalah

Permasalahan yang dibahas dalam Tugas Akhir memiliki beberapa batasan, yakni sebagai berikut :

1. Data penelitian yang diambil merupakan data Tugas Akhir untuk program studi S1 dari Pusat Data Lembaga Pengembangan Teknologi dan Sistem Informasi (LPTSI) ITS.
2. Pembaharuan data kemiripan peneliti dilakukan manual oleh administrator secara berkala.
3. Kakas yang dibangun menggunakan bahasa pemrograman PHP dengan kerangka kerja CodeIgniter, kerangka kerja pengolahan kata Zend Search Lucene Framework, kerangka kerja visualisasi graf berbasis *web* Vis.js, dan basis data SQL Server 2005.

1.4 Tujuan

Tujuan dari pembuatan Tugas Akhir ini adalah untuk merancang dan membangun sebuah kakas yang akan menjadi bagian dari fitur Sistem Repositori Peneliti. Kakas tersebut berupa visualisasi peta kerjasama peneliti berdasarkan topik penelitian yang ada di ITS menggunakan pemodelan graf.

1.5 Manfaat

Adapun manfaat dari Tugas Akhir ini adalah untuk memudahkan para peneliti, mahasiswa dan masyarakat umum mengetahui peta kerjasama peneliti yang ada di ITS berdasarkan kemiripan topik penelitiannya.

1.6 Metodologi

Langkah yang ditempuh dalam pengerjaan Tugas Akhir ini adalah sebagai berikut.

1. Studi Literatur

Mengumpulkan literatur yang dibutuhkan dalam proses perancangan dan implementasi kakas yang akan dibangun. Adapun tahap studi literatur antara lain sebagai berikut :

- a. Penggunaan kerangka kerja bahasa pemrograman PHP yaitu CodeIgniter.
- b. Penggunaan kerangka kerja visualisasi pemodelan graf berbasis *web* yaitu Vis.Js.
- c. Penggunaan kerangka kerja pengolahan kata yaitu Zend Search Lucene Framework.
- d. Mempelajari beberapa literatur mengenai metode hubungan antar peneliti berdasarkan penelitian yang sudah ada.

2. Analisis Sistem

Pada tahap ini dilakukan analisa kebutuhan sistem sebagai solusi atas permasalahan yang dihadapi pengguna. Langkah pada tahap ini antara lain:

- a. Analisa aktor yang terlibat dalam sistem.

- b. Perancangan *use case diagram* sebagai analisa kebutuhan fungsional sistem.
- c. Analisa kebutuhan non-fungsional.
- d. Analisa arsitektur sistem.

3. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem dari hasil analisa terhadap sistem. Proses analisa digambarkan dalam bentuk diagram atau bagan untuk mempermudah gambaran rancangan sistem. Langkah perancangan pada tahap ini antara lain:

- a. Rancangan basis data sistem.
- b. Rancangan visualisasi peta kerjasama peneliti.
- c. Rancangan manajemen penghitungan bobot hubungan antar peneliti.
- d. Rancangan antarmuka kakas web visualisasi peta kerjasama peneliti dan manajemen penghitungan bobot hubungan antar peneliti.

4. Implementasi

Pada tahap ini dilakukan pembuatan kakas berdasarkan rancangan yang telah dibuat pada proses sebelumnya. Rincian pada tahap ini sebagai berikut :

- a. Implementasi metode menghitung hubungan peneliti berdasarkan penelitian yang pernah dibuat.
- b. Implementasi rancangan visualisasi peta kerjasama peneliti.
- c. Implementasi rancangan manajemen penghitungan bobot hubungan antar peneliti.
- d. Implementasi rancangan basis data.
- e. Implementasi rancangan antarmuka visualisasi peta kerjasama dan manajemen bobot antar peneliti.

5. Pengujian dan evaluasi

Pada tahap ini dilakukan pengujian dan evaluasi terhadap kakas. Tahap ini dilakukan dalam beberapa cara, yaitu:

- 1. Pengujian *black box*
Pengujian *black box* adalah pengujian yang berfokus pada spesifikasi fungsional perangkat lunak.

2. Pengujian usabilitas

Pengujian fungsional dilakukan dengan cara melakukan survei pengguna yaitu dosen/peneliti, mahasiswa dan masyarakat umum.

6. Penyusunan Buku Tugas Akhir

Pada tahap ini ditulis buku yang bertujuan untuk mendokumentasikan seluruh konsep, rancangan, dasar teori, literatur, proses yang dilakukan dan hasil yang diperoleh selama pengerjaan tugas akhir. Buku yang ditulis bertujuan untuk memberikan gambaran dari pengerjaan tugas akhir dan berguna untuk pembaca yang tertarik untuk melakukan pengembangan sistem lebih lanjut.

1.7 Sistematika

Pendokumentasian seluruh konsep, rancangan, dasar teori, literature, proses yang dilakukan dan hasil yang diperoleh selama pengerjaan tugas akhir. Buku yang ditulis bertujuan untuk memberikan gambaran dari pengerjaan tugas akhir dan berguna untuk pembaca yang tertarik untuk melakukan pengembangan sistem lebih lanjut.

Buku tugas akhir akan terdiri dari beberapa bagian, yaitu:

Bab I Pendahuluan

Bab ini membahas latar belakang masalah, tujuan pembuatan tugas akhir, rumusan permasalahan, batasan permasalahan, manfaat tugas akhir, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

Bab II Tinjauan Pustaka

Bab ini membahas teori pendukung dan literatur yang berkaitan dengan bahasan dan mendasari pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan

Bab ini membahas tentang desain dan rancangan dari perangkat lunak. Rancangan dan desain meliputi data, proses, arsitektur.

Bab IV Implementasi

Bab ini membahas tentang implementasi hasil analisis dan perancangan dalam bentuk *coding*. Bab ini membahas proses pembangunan kakas.

Bab V Pengujian dan Evaluasi

Bab ini membahas tentang pengujian aplikasi berdasarkan skenario yang telah ditentukan. Mengevaluasi fitur aplikasi apakah telah memenuhi kebutuhan fungsional.

Bab V Kesimpulan dan Saran

Bab ini berisi kesimpulan baik dari proses pengembangan perangkat lunak dan hasil uji coba.

Daftar Pustaka

Merupakan daftar referensi yang digunakan dalam pembuatan tugas akhir.

Lampiran

Merupakan bab tambahan yang berisi kode–kode sumber yang penting pada aplikasi.

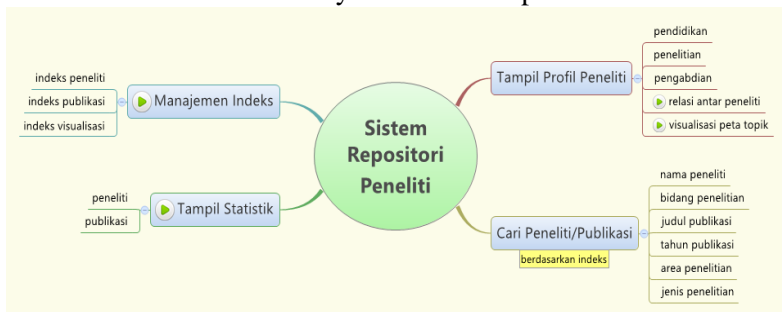
BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dibahas mengenai dasar teori dan literatur yang menjadi dasar pembuatan Tugas Akhir.

2.1 Sistem Repositori Peneliti Institut Teknologi Sepuluh Nopember (ITS) Surabaya

Perguruan Tinggi merupakan salah satu lembaga penelitian yang paling berpengaruh terhadap kemajuan dunia penelitian disuatu negara. Hal ini karena banyaknya penyumbang publikasi penelitian berasal dari Perguruan Tinggi. Saat ini untuk menggiatkan dunia penelitian di Indonesia, Institut Teknologi Sepuluh Nopember (ITS) Surabaya sedang mengembangkan sebuah sistem informasi yang akan menunjang dunia penelitian khususnya di lingkungan ITS. Sistem informasi tersebut yaitu Sistem Repositori Peneliti.



Gambar 2. 1 Fitur Sistem Repositori Peneliti

Sistem Repositori Peneliti atau disebut juga dengan Resources ITS (RESITS) nantinya akan menjadi versi terbaru dari Sistem Informasi Personal Dosen yang sebelumnya sudah pernah ada [3]. Sistem Informasi Repositori Peneliti secara

husus akan memberikan informasi kepada masyarakat seputar dunia penelitian yang ada di ITS. Adapun fitur-fitur pada Sistem Repositori Peneliti terdapat pada Gambar 2. 1.

Adapun keterangan untuk setiap fitur Sistem Repositori Peneliti adalah sebagai berikut:

1. Manajemen Indeks

Indeks merupakan semacam kamus data yang akan digunakan pada mesin pencari yang akan dibuat, yaitu tempat dimana dokumen-dokumen diekstrak informasi (misal: tautan, judul, isi dokumen) kemudian diurutkan menjadi satu kesatuan yang terindeks.

Manajemen Indeks digunakan untuk mengelola (secara manual) indeks yang dihasilkan dari kamus data. Indeks tersebut dihasilkan menggunakan *library* Zend Lucene Framework. Data yang didapatkan dari kamus data disimpan kedalam bentuk *file binary*. Kemudian *file binary* tersebut akan digunakan dalam fitur pencarian Sistem Repositori Peneliti. Hal ini dikarenakan kriteria pencarian dalam sistem tersebut pada kamus data berasal dari beberapa tabel yang berbeda, sehingga jika dilakukan *query* konvensional pada setiap pencarian maka akan memakan sumber daya yang cukup besar. Terdapat dua macam indeks yang digunakan pada sistem tersebut, indeks peneliti menyimpan data peneliti dan indeks publikasi menyimpan data publikasi ilmiah peneliti. Indeks dapat diperbarui secara manual sesuai keperluan agar data yang ditampilkan sesuai dengan kamus data.

2. Tampil Statistik

Tampil Statistik digunakan untuk menampilkan statistik dari peneliti dan publikasi berdasarkan area peneliti (fakultas).

3. Tampil Profil Peneliti
Tampil Profil Peneliti digunakan untuk melihat data dari setiap peneliti, seperti pendidikan, pengabdian, publikasi ilmiah, visualisasi peta topik, relasi antar peneliti (peneliti yang sering bekerja sama), dll.
4. Cari Peneliti/Publikasi
Cari Peneliti/Publikasi digunakan untuk melakukan pencarian peneliti ataupun publikasi dengan kriteria tertentu, yaitu nama peneliti, bidang penelitian, tahun publikasi, area penelitian, tahun publikasi serta jenis penelitian.

2.2 Web Personal Dosen Institut Teknologi Sepuluh Nopember (ITS) Surabaya

Web Personal Dosen ITS merupakan sebuah web yang menyediakan informasi kepada masyarakat umum tentang dosen-dosen yang ada di ITS. Sebagai masyarakat umum, pengguna dapat menemukan berbagai informasi terkait dosen-dosen ITS, seperti biodata personal, publikasi ilmiah, bidang riset, dan bidang penelitian. Pengguna juga dapat menelusuri data-data dosen yang telah dikelompokkan baik per fakultas maupun per jurusan.

Sedangkan sebagai dosen, ini merupakan sarana untuk mempublikasikan publikasi ilmiah, seminar, buku, dan hasil penelitian lainnya. Disini dosen juga dapat menampilkan informasi akademiknya maupun informasi biodata diri yang lain [3]. Adapun tampilan Web Personal Dosen dapat dilihat pada Gambar 2. 2.



Gambar 2. 2 Tampilan Web Personal Dosen ITS

2.3 CodeIgniter

CodeIgniter merupakan kerangka kerja aplikasi web yang bersifat terbuka. Kerangka kerja ini digunakan untuk membangun situs web PHP yang dinamis. Kerangka kerja ini menyediakan berbagai macam *library* untuk kebutuhan pada umumnya, racangan antarmuka yang sederhana dan struktur yang logis untuk mengakses *library* yang telah disediakan.

Logo CodeIgniter dapat dilihat pada Gambar 2. 3. Kerangka kerja CodeIgniter menggunakan Model – View – Controller. Dalam kerangka kerja CodeIgniter, *controller* merupakan bagian yang penting dalam mengembangkan situs

web menggunakan kerangka kerja ini, sedangkan model adalah hal yang opsional. Selain itu kerangka kerja CodeIgniter juga terdokumentasi dengan baik dan lengkap sehingga memudahkan pengembang untuk mengembangkan situs web menggunakan kerangka kerja tersebut [4].



Gambar 2. 3 Logo CodeIgniter

2.4 Zend Search Lucene Framework Library

Zend Framework merupakan salah satu kakas kerja bahasa pemrograman PHP yang berbasis *OOP (Object Oriented Programming)* serta bersifat *open source*. Fokus utama dari kakas kerja ini adalah untuk membangun aplikasi *web* dan *web service* yang lebih aman, reliabel, dan modern. Selain itu juga telah didukung *API* untuk beberapa vendor seperti Google, Amazon, Yahoo!, Flickr.

Beberapa fitur terbaru Zend Framework antara lain: dukungan *AJAX* melalui *JSON*, Search-Lucene, Syndication, Web Services, dan OOP PHP5.



Gambar 2. 4. Logo Zend Framework

Adapun logo Zend Framework dapat dilihat pada Gambar 2. 4. Zend Framework menyediakan sebuah fitur *Zend_Search_Lucene* yang berguna untuk pencarian *full text* pada dokumen. Fitur ini merupakan adopsi dari *project Apache Lucene*, yang merupakan *project full-featured full text* mesin pencari yang dibangun dalam bahasa Java.

Secara garis besar, dalam pembuatan mesin pencari berbasis *web* yang diperlukan adalah:

1. meng-*install* Zend Framework,
2. melakukan pengindeksan terhadap dokumen-dokumen yang akan digunakan sebagai kamus data pencarian,
3. melakukan pencarian dalam indeks yang telah dibentuk.

Indeks merupakan semacam kamus data yang akan digunakan pada mesin pencari yang akan dibuat, yaitu tempat dimana dokumen-dokumen diekstrak informasi (misal: tautan, judul, isi dokumen) kemudian diurutkan menjadi satu kesatuan yang terindeks.

Zend Framework juga memberikan fitur pencarian menggunakan *query parser*, dimana dengan *query parser* hasil pencarian dapat disesuaikan dengan keinginan, misal menambahkan operator *AND*, *OR*, dll [5].

2.5 Vis.Js



Gambar 2. 5. Logo Vis.Js

Vis.Js merupakan kerangka kerja visualisasi berbasis web. Kerangka kerja ini dibangun sedemikian rupa sehingga mudah digunakan, dapat menangani data dinamis dalam jumlah besar, serta dapat memanipulasi data dan interaksi dengan data. Kerangka kerja ini terdiri dari komponen *DataSet*, *Timeline*, *Graph*, dan *Graph 3d*.

Logo Vis.Js dapat dilihat pada Gambar 2. 5. Vis.Js dikembangkan oleh Almende B.V, yang merupakan bagian dari CHAP. Vis.Js dapat berjalan baik pada Chrome, Firefox, Opera, Safari, IE9+, dan kebanyakan *mobile browsers* [6].

Kerangka kerja ini merupakan bahasa pemrograman yang memberikan kemampuan tambahan terhadap bahasa HTML dengan memungkinkan pengekseskuan perintah perintah di sisi pengguna (*web client*). Kerangka kerja ini dapat memodelkan *dataset* dengan pemodelan graf berbasis web.

2.6 JavaScript

Javascript diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser

Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru-buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali kita membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman “Java”) pada masa itu, maka Netscape memberikan nama “JavaScript” kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “Jscript” di browser Internet Explorer 3.

Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web.

Javascript bergantung kepada *browser* yang memanggil halaman web yang berisi skrip-skrip dari Javascript dan tentu saja terselip di dalam dokumen HTML. Javascript juga tidak memerlukan kompilator atau penterjemah khusus untuk menjalankannya (pada kenyataannya kompilator Javascript sendiri sudah termasuk di dalam *browser* tersebut). Lain halnya dengan bahasa Java (yang mana JavaScript selalu di banding-bandingkan) yang memerlukan kompilator khusus untuk menterjemahkannya di sisi user/klien [7].

2.7 JavaScript Object Notation

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan. JSON mudah untuk dibaca dan ditulis oleh manusia dan memudahkan mesin untuk mengurai dan menciptakan. Format ini berdasar dari himpunan bagian bahasa pemrograman JavaScript standar ECMA-262 edisi ke-3, Desember 1999.

Logo JSON dapat dilihat pada Gambar 2. 6. JSON merupakan format teks yang benar – benar berbeda atau independen namun tetap menggunakan konvensi yang mudah dikenali bagi programmer yang sudah terbiasa dengan bahasa C, C++, C#, Java, JavaScript, Perl, Python dan bahasa lainnya. Sifat JSON yang seperti ini menjadikannya format pertukaran data yang ideal [8].



Gambar 2. 6 Logo JSON

2.8 Algoritma TF-IDF (Term Frequency - Inversed Document Frequency)

Algoritma *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah suatu metode pemberian bobot hubungan suatu *term* (kata) terhadap dokumen. Metode ini menggabungkan dua konsep untuk perhitungan bobot, yaitu *Term Frequency* (TF) merupakan frekuensi kemunculan *term* (t) pada setiap dokumen (d), dan *Inverse Document Frequency* (IDF) merupakan nilai bobot suatu *term* dihitung dari seringnya *term* muncul dibeberapa dokumen.

Pada metode ini pembobotan kata dalam sebuah dokumen dilakukan dengan mengalikan nilai TF dan IDF. Pembobotan diperoleh berdasarkan jumlah kemunculan *term* dalam dokumen (TF) dan jumlah kemunculan *term* pada seluruh dokumen (IDF). Bobot suatu *term* semakin besar jika *term* tersebut sering muncul dalam suatu dokumen dan semakin kecil jika *term* tersebut muncul dalam banyak dokumen [9]. Bobot TF IDF sebuah *term* pada dokumen tertentu dapat dihitung menggunakan Persamaan 2.1.

$$TF\ IDF(t_k, d_j) = TF(t_k, d_j) \bullet IDF(t_k) \quad (2.1)$$

dengan,

t_k = *term* ke- k

d_j = dokumen ke- j

$TF(t_k, d_j)$ = *term frequency term* ke- k pada dokumen ke- j

$IDF(t_k)$ = *inverse document frequency term* ke- k

$TF\ IDF(t_k, d_j)$ = bobot *term* ke- k pada dokumen ke- j

Untuk menghitung bobot IDF setiap *term*, digunakan Persamaan 2.2.

$$IDF(t_k) = \log \frac{N}{df(t)} \quad (2.2)$$

dengan,

N = jumlah dokumen

$df(t)$ = banyaknya dokumen yang mengandung *term* (t)

$IDF(t_k)$ = *inverse document frequency term* ke- k

Untuk mendapatkan TF setiap dokumen digunakan Persamaan 2.3.

$$TF(t_k, d_j) = f(t_k, d_j) \quad (2.3)$$

dengan,

$TF(t_k, d_j)$ = *term frequency term* ke- k pada dokumen ke- j
 $f(t_k, d_j)$ = frekuensi munculnya *term* ke- k pada dokumen ke- j

Contoh implementasi sederhana dari TF-IDF adalah sebagai berikut :

Kata kunci (kk) = pengetahuan logistik
 Dokumen 1 (D₁) = manajemen transaksi logistik
 Dokumen 2 (D₂) = pengetahuan antar individu
 Dokumen 3 (D₃) = dalam manajemen pengetahuan
 terdapat transfer pengetahuan logistik

Jadi jumlah dokumen (N) = 3

Setelah dilakukan tahap *tokenizing* dan proses *filtering*, maka kata **antar** pada dokumen 2 serta kata **dalam** dan **terdapat** pada dokumen 3 dihapus. *Tokenizing* adalah pemotongan string input berdasarkan tiap kata yang menyusunnya. *Filtering* adalah mengambil kata-kata penting dari hasil token. Perhitungan TF-IDF dapat dilihat pada Tabel 2. 1.

Tabel 2. 1 Contoh Perhitungan TF-IDF

Token	tf				d f	D/ df	IDF	TF-IDF			
	kk	D1	D2	D3				kk	D1	D2	D3
manajemen	0	1	0	1	2	1.5	0.176	0	0.176	0	0.176
transaksi	0	1	0	0	1	3	0.477	0	0.477	0	0

logistik	1	1	0	1	2	1.5	0.176	0.176	0.176	0	0.176
transfer	0	0	0	1	1	3	0.477	0	0	0	0.477
pengetahuan	1	0	1	2	2	1.5	0.176	0.176	0	0.176	0.352
individu	0	0	1	0	1	3	0.477	0	0	0.477	0
Total								0.352	0.829	0.653	1.181

TF-IDF untuk $D_1 = 0.176 + 0 = 0.176$

TF-IDF untuk $D_2 = 0 + 0.176 = 0.176$

TF-IDF untuk $D_3 = 0.176 + 0.352 = 0.528$

Dari contoh studi kasus di atas, dapat diketahui bahwa bobot TF-IDF dari D_1 dan D_2 adalah sama. Apabila hasil pengurutan bobot dokumen tidak dapat mengurutkan secara tepat, karena bobot keduanya sama, maka diperlukan proses perhitungan dengan algoritma *vector space model*. Yaitu dengan menghitung nilai *cosinus* sudut dari dua vektor, yaitu TF-IDF dari tiap dokumen dan TF-IDF dari kata kunci.

2.9 Algoritma Vector Space Model

Vector Space Model adalah suatu model yang digunakan untuk mengukur kemiripan antara suatu dokumen dengan suatu *query*. Pada model ini, *query* dan dokumen dianggap sebagai vektor-vektor pada ruang n -dimensi, dimana n adalah jumlah dari seluruh *term* yang ada dalam *leksikon*. *Leksikon* adalah daftar semua *term* yang ada dalam indeks. Salah satu cara untuk mengatasi hal tersebut dalam model *vector space* adalah dengan cara melakukan perluasan vektor. Proses perluasan dapat dilakukan pada vektor *query*, vektor dokumen, atau pada kedua vektor tersebut.

Pada algoritma *vector space model* menggunakan rumus untuk mencari nilai *cosinus* sudut antara dua vektor dari setiap bobot TF-IDF dokumen (WD) dan bobot TF-IDF dari kata kunci (WK). Rumus yang digunakan terdapat pada Persamaan 2.4.

$$\cos \theta \rightarrow \text{sim}(d_j, q) = \frac{d_j * q}{\|d_j\| \times \|q\|} = \frac{\sum_{i=1}^N w_{ij} \times w_{iq}}{\sqrt{\sum_{i=1}^N w_{ij}^2} \times \sqrt{\sum_{i=1}^N w_{iq}^2}} \quad (2.4)$$

dengan,

$\|d_j\|$ = vektor dokumen ke- j

$\|q\|$ = vektor kata kunci (*query*)

W_{ij} = bobot TF-IDF dokumen ke- j term ke- i

W_{iq} = bobot TF-IDF kata kunci (*query*) term ke- i

$\text{sim}(d_j, q)$ = nilai similaritas antara dokumen ke- j dengan kata kunci (*query*)

Apabila studi kasus pada algoritma TF-IDF pada Tabel 2.1 dicari nilai *cosinus* sudut antara vektor masing-masing dokumen dengan kata kunci, maka hasil yang didapatkan akan lebih presisi. Seperti yang ditunjukkan pada Tabel 2. 3.

Tabel 2. 2 Perhitungan Vector Space Model

Token	kk	D1	D2	D3	kk*D1	kk*D2	kk*D3
manajemen	0	0.031	0	0.031	0	0	0
transaksi	0	0.228	0	0	0	0	0
logistik	0.031	0.031	0	0.031	0.031	0	0.031
transfer	0	0	0	0.228	0	0	0
pengetahuan	0.031	0	0.031	0.124	0	0.031	0.062
individu	0	0	0.228	0	0	0	0
sqrt(kk)		sqrt(Di)			sqrt(kk*Di)		
0.249		0.539	0.509	0.643	0.031	0.031	0.093

Selanjutnya menghitung nilai cosinus sudut antara vektor kata kunci dengan tiap dokumen dengan menggunakan Persamaan 2.3.

$$\text{Cosine}(D_i) = \text{sum}(kk.D_i) / (\text{sqrt}(kk) * \text{sqrt}(D_i))$$

- Untuk Dokumen 1 (D_1)

$$\begin{aligned}\text{Cosine}(D_1) &= \text{sum}(kk.D_1) / (\text{sqrt}(kk) * \text{sqrt}(D_1)) \\ &= 0.031 / (0.249 * 0.539) \\ &= 0.231\end{aligned}$$
- Untuk Dokumen 2 (D_2)

$$\begin{aligned}\text{Cosine}(D_2) &= \text{sum}(kk.D_2) / (\text{sqrt}(kk) * \text{sqrt}(D_2)) \\ &= 0.031 / (0.249 * 0.509) \\ &= 0.245\end{aligned}$$
- Untuk Dokumen 3 (D_3)

$$\begin{aligned}\text{Cosine}(D_3) &= \text{sum}(kk.D_3) / (\text{sqrt}(kk) * \text{sqrt}(D_3)) \\ &= 0.093 / (0.249 * 0.643) \\ &= 0.581\end{aligned}$$

Tabel 2. 3 Tabel hasil nilai cosinus setiap dokumen

	D1	D2	D3
<i>cosine</i>	0.231	0.245	0.581
	Peringkat 3	Peringkat 2	Peringkat 1

Sesuai perhitungan di atas maka nilai *cosinus* setiap dokumen telah didapat, seperti yang ditunjukkan pada Tabel 2. 3.

Dengan hasil akhir tersebut dapat diketahui bahwa dokumen 3 (D_3) memiliki tingkat similaritas tertinggi terhadap kata kunci, kemudian disusul dengan D_2 dan D_1 .

2.10 Algoritma Dice's coefficient

Dice's coefficient digunakan untuk menghitung persamaan dari dua sampel data. Persamaan *Dice's coefficient* dapat dilihat pada Persamaan 2.5.

$$S = \frac{2|X \cap Y|}{|X| + |Y|} \quad (2.5)$$

dengan,

$|X \cap Y|$ = jumlah yang dikerjakan X dan Y

$|X|$ = jumlah yang dikerjakan X

$|Y|$ = jumlah yang dikerjakan Y

Contoh

$X = \{aa, bb, cc, dd\}$

$Y = \{ee, ff, gg, dd\}$

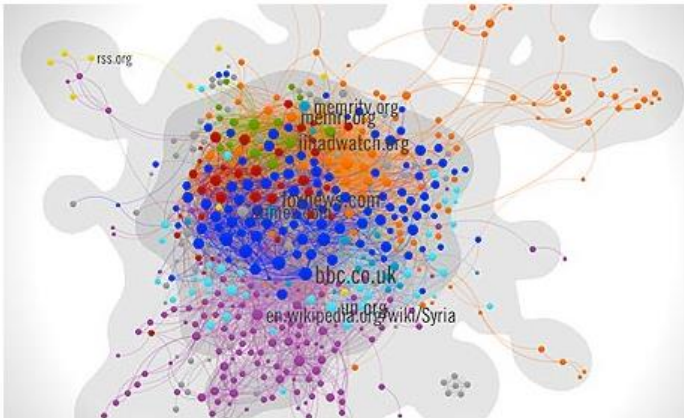
$S = (2 \cdot 1) / (4 + 4) = 0.25$

2.11 Referensi aplikasi yang sudah pernah ada

2.11.1 Mapping Breivik's mind

Mapping Breivik's mind merupakan sebuah hasil analisa *Anders Behring Breivik* tentang pandangannya yang meliputi *Islamophobia*, dukungan untuk *zionis* sayap kanan, dan feminisme. Karena ketakutannya terhadap tiga hal tersebut, dia membuat sebuah analisis tentang penyebaran hubungan situs-situs yang ada di Eropa dan Amerika dan berkaitan dengan tiga hal tersebut. *Mapping Breivik's mind* dikembangkan oleh pengembang asal Perancis yaitu *LinkFluence* [10]. Perusahaan ini fokus pada pengembangan kakas yang berhubungan dengan *social network*. Dalam pengembangan *Mapping Breivik's mind* ini, *LinkFluence* menggunakan *library* tambahan *SigmaJs* untuk

memvisualisasikan hasil analisa *Anders Behring Breivik* dalam bentuk pemodelan graf.



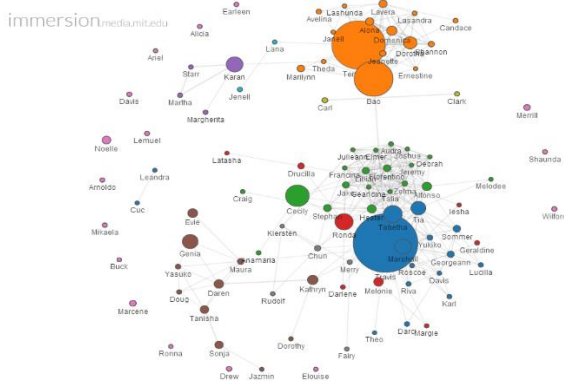
Gambar 2. 7 Peta Mapping Breivik's mind

Mapping Breivik's mind dapat dilihat pada Gambar 2. 7. Peta tersebut menunjukkan hubungan-hubungan antar situs. Setiap *node* pada peta tersebut merepresentasikan sebuah situs. Hubungan antar situs direpresentasikan oleh *edge* yang menghubungkan antar *node*. Situs-situs yang memiliki hubungan direpresentasikan dengan warna *edge* yang berbeda-beda. Kemudian pada saat salah satu *node* dipilih maka akan muncul situs mana saja yang berhubungan dengan *node* yang dipilih.

2.11.2 Immersion

Immersion merupakan visualisasi pemetaan dari sekumpulan *email* yang sering berinteraksi. Visualisasi pemetaan yang digunakan pada *Immersion* menggunakan pemodelan jaringan berbentuk graf. Setiap *email*

direpresentasikan dengan lingkaran (*node*) dengan warna dan ukuran berbeda-beda yang menunjukkan karakteristik masing-masing *email*. Kemudian besarnya interaksi antar *node* direpresentasikan dengan garis (*edge*). Ini merupakan salah satu perspektif yang berbeda dari sebuah *email*. *Email* seseorang dilihat dari riwayat pertama kali *email* dibuat hingga dengan *email* siapa saja *email* seseorang tersebut berinteraksi. Semakin sering seseorang berinteraksi, seperti mengirim atau menerima *email*, maka jarak kedua *email* tersebut semakin dekat. Salah satu harapan yang ingin dicapai oleh *Immersion* yaitu seseorang dapat memprediksi hubungan profesional dirinya dengan orang lain di masa depan. Contoh pemetaan *email Immersion* dapat dilihat pada Gambar 2. 8.



Gambar 2. 8 Contoh Pemetaan Email Immersion

BAB III

ANALISIS DAN PERANCANGAN

Pada bab ini dibahas analisa permasalahan, analisa kebutuhan, dan perancangan kakas dalam Tugas Akhir. Bagian awal bab akan dibahas tentang analisa permasalahan dan kebutuhan masyarakat. Berikutnya dibahas fungsional sistem yang berdasarkan hasil analisa kebutuhan. Bagian berikutnya akan dibahas rancangan perangkat lunak yang ditujukan untuk memberikan gambaran tentang perangkat lunak yang dibuat.

3.1. Analisis

Pada sub bab berikut akan dijelaskan analisa pembuatan kakas visualisasi peta kerjasama peneliti. Analisa yang dilakukan meliputi analisa permasalahan, kebutuhan umum perangkat lunak, deskripsi umum sistem, arsitektur dan kebutuhan fungsional sistem.

3.1.1. Permasalahan

Perkembangan ilmu pengetahuan dan teknologi semakin berkembang pesat. Bila suatu negara ingin bersaing di dunia global, maka kemampuan untuk menguasai ilmu pengetahuan dan teknologi menjadi wajib dimiliki oleh negara tersebut. Maka dari itu, penelitian dan pengembangan harus menjadi perhatian penting sebagai kunci kemajuan suatu negara.

Persaingan di dunia global salah satunya dapat dilakukan dengan mendapatkan peluang-peluang topik penelitian baru yang nantinya akan berguna di masa depan. Peluang ini bisa didapatkan, salah satunya dengan melakukan identifikasi bidang utama dari disiplin ilmu dan publikasi penelitian yang sudah ada saat ini. Dari identifikasi tersebut nantinya dapat dipetakan adanya hubungan dari beberapa elemen penelitian seperti peneliti, publikasi ilmiah serta topik penelitian. Adapun peluang-peluang penelitian yang

mungkin muncul dari hubungan tersebut seperti, peluang kerjasama antar peneliti dengan topik penelitian yang sama, peluang topik penelitian baru yang mungkin ada, serta peluang-peluang penelitian lainnya. Namun hingga saat ini, di Indonesia masih belum banyak lembaga penelitian yang memberikan sajian informasi mengenai pemetaan penelitian ini.

Perguruan Tinggi merupakan salah satu lembaga penelitian yang paling berpengaruh terhadap kemajuan dunia penelitian disuatu negara. Hal ini karena banyaknya penyumbang publikasi penelitian berasal dari Perguruan Tinggi. Institut Teknologi Sepuluh Nopember (ITS) Surabaya merupakan salah satu Perguruan Tinggi yang aktif memberikan kontribusi di dunia penelitian di Indonesia. Saat ini dari peneliti-peneliti di ITS, tentunya sudah banyak aktivitas penelitian yang dilakukan, salah satunya penelitian tugas akhir. Aktivitas penelitian tugas akhir tersebut setiap tahunnya kian bertambah dan menghasilkan dokumen penelitian tugas akhir yang tersimpan dalam sebuah basis data. Sebenarnya, hubungan kerjasama penelitian antar peneliti bisa didapatkan dari dokumen penelitian yang tersimpan dalam basis data tersebut. Namun hingga saat ini masih belum ada representasi visual yang merepresentasikan hubungan-hubungan kerjasama antar peneliti tersebut.

Dari dokumen penelitian tugas akhir yang tersimpan, dapat dilakukan suatu pengolahan dokumen tugas akhir untuk mendapatkan keterkaitan antar peneliti. Misalnya saja, semakin sering peneliti-peneliti melakukan bimbingan bersama terhadap penelitian tugas akhir, maka hubungan kerjasama antar peneliti tersebut semakin dekat. Selain itu, juga bisa didapatkan keterkaitan antar peneliti berdasarkan topik penelitian berdasarkan hasil pengolahan ekstraksi dokumen tugas akhir. Dari hubungan-hubungan antar peneliti tersebut dapat dilakukan pemetaan kerjasama peneliti, yaitu suatu bentuk representasi visual hubungan-hubungan kerjasama penelitian dari seluruh peneliti yang ada. Representasi visual hubungan kerjasama antar peneliti

merupakan sebuah inovasi untuk memudahkan para peneliti maupun masyarakat mengetahui adanya hubungan kerjasama antar peneliti. Selain itu, peluang-peluang kerjasama penelitian baru juga bisa didapatkan dari representasi visual hubungan kerjasama antar peneliti tersebut.

3.1.2. Deskripsi Umum Perangkat Lunak

Kakas yang dibangun dalam Tugas Akhir ini merupakan kakas visualisasi pemetaan kerjasama peneliti berdasarkan topik menggunakan pemodelan graf. Kakas visualisasi ini dapat memberikan informasi kemiripan dosen berdasarkan penelitian yang pernah dibuat. Namun dalam Tugas Akhir ini, data penelitian yang digunakan masih merupakan data yang diambil dari Tugas Akhir Program Studi S1 Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Dalam pengembangan selanjutnya sangat dimungkinkan untuk menambahkan jenis dokumen yang lain seperti jurnal, seminar nasional, seminar internasional, dan lain sebagainya.

Kakas visualisasi ini menyajikan informasi peta kerjasama peneliti di ITS. Visualisasi peta kerjasama peneliti yang ditampilkan diolah dari dokumen tugas akhir dimana peneliti tersebut menjadi pembimbingnya. Hubungan kerjasama peneliti diolah dari setiap dokumen tugas akhir diambil judul, abstraksi, dan kata kunci tugas akhir yang pernah dibimbing oleh peneliti bersangkutan.

Selanjutnya dari judul, abstraksi, dan kata kunci tugas akhir diolah sehingga didapatkan kata (*term*) yang penting saja. Untuk setiap peneliti dengan seluruh dokumen tugas akhir yang pernah dibimbingnya akan didapatkan beberapa *term* terpenting teratas yang dianggap sebagai topik penelitian peneliti tersebut. Kemudian dari beberapa *term* terpenting setiap peneliti didapatkan hubungan antar peneliti berdasarkan perhitungan banyaknya *term* penting yang dipakai bersama. Semakin banyak *term* dipakai bersama maka kedua peneliti tersebut dianggap semakin mirip.

Selain itu, hubungan kerjasama antar peneliti juga dipengaruhi oleh banyaknya kerjasama para peneliti dalam satu tugas akhir. Semakin sering para peneliti bekerjasama dalam satu tugas akhir maka hubungan kerjasamanya semakin dekat.

Hubungan antar peneliti yang sudah terbentuk kemudian divisualisasikan menggunakan pemodelan graf berbasis *web*. Dari visualisasi tersebut, pengguna dapat melihat peneliti-peneliti yang mirip topik penelitiannya untuk setiap peneliti. Selain itu pengguna juga dapat melihat detail informasi untuk setiap peneliti melalui tautan yang sudah tersedia. Kemudian selain pengguna dapat melihat peta kerjasama peneliti seluruh ITS, pengguna juga dapat melihat peta kerjasama berdasarkan area penelitian (fakultas) tertentu.

Kakas visualisasi peta kerjasama antar peneliti dibangun dengan menggunakan SQL Server, JSON, serta beberapa kerangka kerja seperti Zend Search Lucene, Vis.js, dan CodeIgniter PHP Framework. SQL Server digunakan untuk menyimpan data penelitian tugas akhir, data peneliti, *term* setiap peneliti, bobot kemiripan antar peneliti, serta perhitungan sementara untuk mendapatkan bobot kemiripan antar peneliti. JSON digunakan untuk menyimpan data peneliti dan bobot kemiripan antar peneliti yang didapatkan dari kamus data. Data yang berasal dari *file* JSON inilah yang akan divisualisasikan kepada pengguna. Zend Search Lucene digunakan untuk mendapatkan daftar *term* yang unik dari keseluruhan dokumen tugas akhir yang diambil dari kamus data, serta untuk menghitung bobot setiap *term*. Vis.js digunakan untuk mevisualisasikan data yang berasal dari *file* JSON dengan visualisasi pemodelan graf berbasis *web*. Sedangkan CodeIgniter merupakan kerangka kerja utama yang digunakan kakas visualisasi ini, terbagi menjadi tiga bagian, Model-Controller-View, *model* digunakan untuk pengkasesan kamus data, *controller* digunakan

untuk memproses suatu data dan atau kontrol terhadap *view*, dan *view* digunakan untuk mengatur tampilan pengguna.

Secara keseluruhan kakas yang dibangun terdiri dari kakas visualisasi itu sendiri serta manajemen visualisasi yang dikelola oleh administrator. Kakas visualisasi menampilkan peta kerjasama peneliti, sedangkan manajemen visualisasi digunakan untuk memperbarui data yang tersimpan di dalam *file* JSON yang diakses oleh kakas visualisasi. Perbaruan data *file* JSON dilakukan secara berkala oleh administrator, terutama ketika ada peneliti atau dokumen penelitian baru.

Fungsionalitas kakas visualisasi peta kerjasama peneliti sebagai berikut.

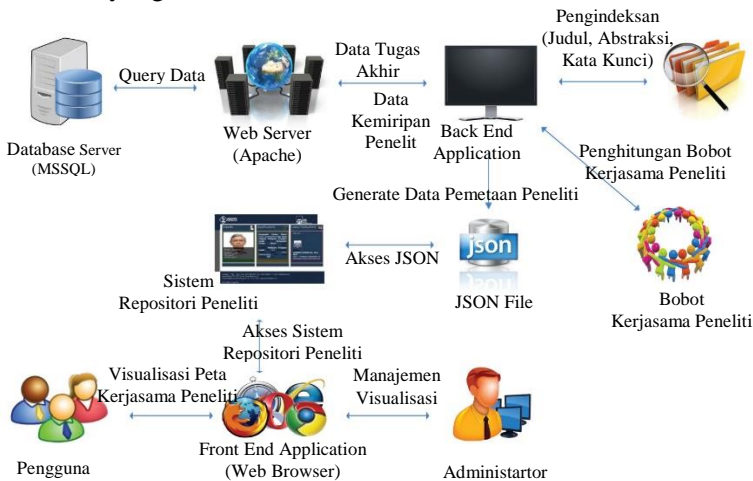
1. Dapat menampilkan peta kerjasama peneliti-peneliti teratas berdasarkan jumlah kemiripan paling banyak dengan peneliti lain.
2. Dapat menampilkan peta kerjasama peneliti berdasarkan fakultas.
3. Dapat melakukan pencarian peneliti.
4. Dapat melihat detail peneliti, meliputi jumlah dokumen penelitian tugas akhir, jumlah peneliti yang mirip topik penelitiannya, jurusan, fakultas serta peneliti-peneliti yang mirip topik penelitiannya.
5. Dapat melihat detail informasi peneliti pada Sistem Repositori Peneliti melalui tautan yang tersedia.

Sedangkan manajemen visualisasi digunakan untuk memperbarui *file* JSON yang berisi data pemetaan penelitian. Pada manajemen visualisasi terdapat beberapa tahapan, tahapan-tahapan tersebut digunakan untuk memperbarui tabel-tabel pada kamus data yang akan digunakan untuk memperbarui data pemetaan peneliti pada *file* JSON.

3.1.3. Arsitektur Sistem

Kakas visualisasi peta kerjasama peneliti menggunakan arsitektur seperti pada Gambar 3. 1. Sedangkan diagram alir kakas

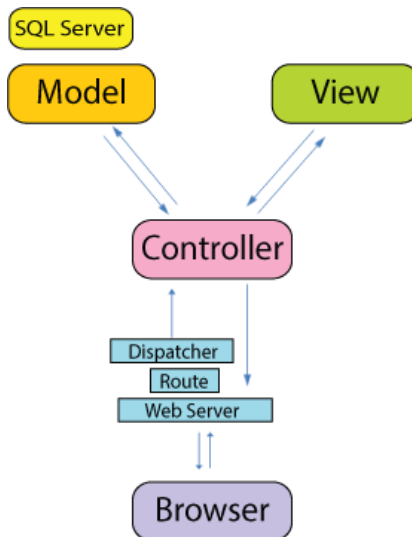
visualisasi dapat dilihat pada Lampiran E tentang diagram alir kakas visualisasi. Kakas ini merupakan salah satu fitur dari Sistem Repositori Peneliti, untuk keperluan visualisasi peta kerjasama peneliti, sistem ini terdapat visualisasi peta kerjasama yang diakses oleh pengguna dan manajemen visualisasi yang dikelola oleh administrator. Kakas pada modul visualisasi menampilkan peta kerjasama peneliti berdasarkan topik penelitian peneliti. Sedangkan kakas pada modul manajemen visualisasi digunakan oleh administrator untuk memperbarui data peta kerjasama peneliti jika terdapat dokumen penelitian maupun peneliti baru, ini dilakukan agar hubungan kerjasama kemiripan topik antar peneliti yang ditampilkan pada modul visualisasi sesuai dengan data penelitian yang berasal dari basis data.



Gambar 3. 1 Arsitektur Sistem

Data peta kerjasama peneliti diolah dari data penelitian tugas akhir yang diambil dari basis data SQL Server menggunakan *web server*. Kemudian pada bagian *back end* aplikasi dilakukan proses pengindeksan data penelitian tugas akhir dan penghitungan bobot

kerjasama peneliti. Setelah didapatkan bobot kerjasama peneliti, bobot kerjasama peneliti tersebut di-generate dalam bentuk *file* notasi graf JSON yang kemudian diakses oleh modul visualisasi. Selanjutnya pada bagian *front end* aplikasi pengguna dapat mengakses modul visualisasi peta kerjasama peneliti menggunakan *web browser*. Dengan menggunakan *web browser* pula administrator mengakses modul manajemen visualisasi untuk memperbarui data peta kerjasama peneliti yang tersimpan dalam *file* JSON. Manajemen visualisasi berisi tahapan-tahapan yang dilakukan pada *back end* aplikasi untuk mengolah data penelitian hingga terbentuk *file* JSON berisi notasi graf peta kerjasama peneliti.



Gambar 3. 2 Control Flow MVC

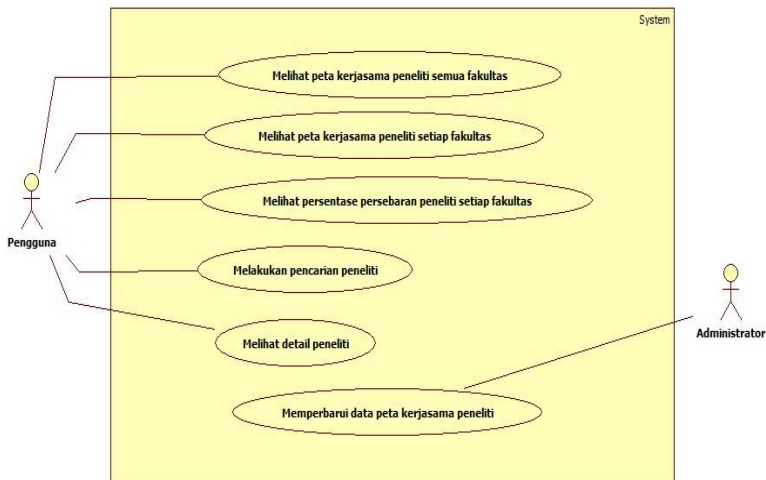
Selain itu berikut gambaran detail penggunaan kerangka kerja CodeIgniter untuk membangun kaskas visualisasi peta kerjasama peneliti berbasis web. CodeIgniter menggunakan konsep MVC

(model-view-controller). *Control flow* arsitektur MVC digambarkan oleh Gambar 3. 2.

3.1.3 Kebutuhan Fungsional

Kakas visualisasi peta kerjasama peneliti menyediakan informasi kepada pengguna tentang peta kerjasama peneliti berdasarkan kemiripan topik penelitiannya. Berikut daftar kebutuhan fungsional dari kakas visualisasi peta kerjasama peneliti termasuk manajemen visualisasinya.

1. Melihat peta kerjasama peneliti semua fakultas.
2. Melihat peta kerjasama peneliti setiap fakultas.
3. Melakukan pencarian peneliti.
4. Melihat persentase persebaran peneliti setiap fakultas.
5. Melihat detail peneliti.
6. Melihat peneliti-peneliti yang mirip topik penelitiannya.
7. Memperbarui data peta kerjasama peneliti.



Gambar 3. 3 Diagram Kasus Visualisasi Peta Kerjasama Peneliti

Kebutuhan fungsional kakas visualisasi peta kerjasama peneiliti dapat dilihat pada Gambar 3. 3. Pengguna dapat melihat peta kerjasama peneliti semua fakultas, melihat peta kerjasama peneliti setiap fakultas, melihat persentase persebaran peneliti setiap fakultas, melakukan pencarian peneliti, melihat detail peneliti dan melihat peneliti yang mirip topik penelitiannya. Administrator dapat memperbarui data peta kerjasama peneliti. Penjelasan lengkap mengenai kasus penggunaan berada di Tabel 3. 1.

Tabel 3. 1 Deskripsi Kasus Penggunaan Visualisasi Peta Kerjasama Peneliti

No	Kode	Nama	Keterangan
1	UC-P01	Melihat peta kerjasama peneliti semua fakultas	Pengguna dapat peta kerjasama peneliti-peneliti teratas yang memiliki kemiripan dengan peneliti lain paling banyak untuk semua fakultas.
2	UC-P02	Melihat peta kerjasama peneliti setiap fakultas	Pengguna dapat peta kerjasama peneliti-peneliti untuk setiap fakultas.
3	UC-P03	Melihat persentase persebaran peneliti setiap fakultas	Pengguna dapat melihat persentase persebaran banyaknya peneliti setiap fakultas.
4	UC-P04	Melakukan pencarian peneliti	Pengguna dapat melakukan pencarian peneliti menggunakan nama peneliti.

5	UC-P05	Melihat detail peneliti	Pengguna dapat melihat detail peneliti seperti nama, jumlah dokumen tugas akhir yang pernah dibimbing, jumlah kemiripan dengan dosen lain pada graf yang sedang aktif, jurusan, fakultas, dan peneliti-peneliti yang mirip topik penelitiannya. Tersedia juga tautan menuju detail peneliti pada Sistem Repositori Peneliti.
6	UC-P06	Memperbarui data peta kerjasama peneliti	Administrator dapat memperbarui data peta kerjasama peneliti.

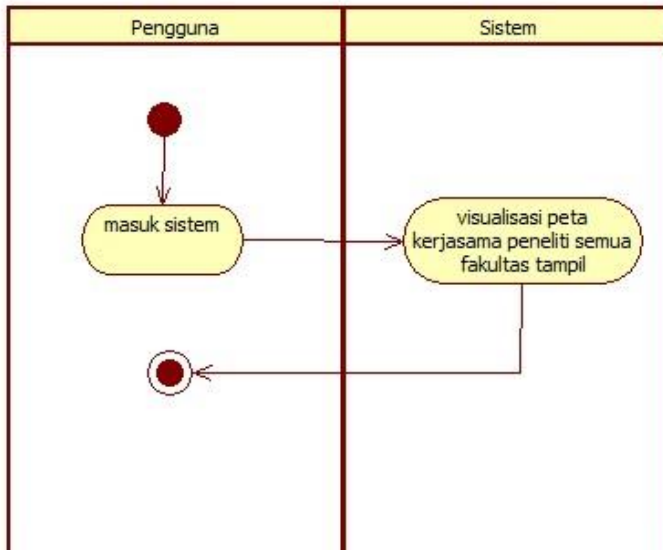
3.1.4.1. Deskripsi Kasus Kebutuhan UC-P01

Kasus kebutuhan kode UC-P01 merupakan kasus kebutuhan melihat peta kerjasama peneliti untuk semua fakultas. Rincian alur kasus Melihat peta kerjasama peneliti semua fakultas dijelaskan pada Tabel 3. 2 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 4.

Tabel 3. 2 Rincian Alur Kasus Penggunaan UC-P01

Nama Case	Use	Melihat peta kerjasama peneliti semua fakultas
Nomor		UC-P01

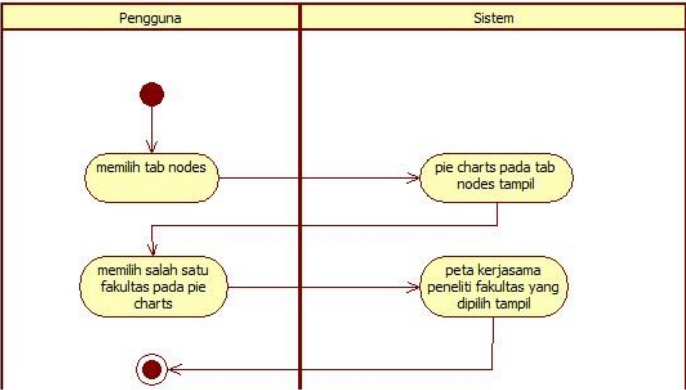
Aktor	Pengguna
Kondisi Awal	Pengguna belum membuka alamat <i>url</i> Sistem Repositori Peneliti untuk fitur visualisasi peta kerjasama peneliti pada <i>web browser</i>
Kondisi akhir	Visualisasi peta kerjasama peneliti untuk semua fakultas tampil
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna menuliskan alamat <i>url</i> fitur visualisasi peta kerjasama peneliti Sistem Repositori Peneliti pada <i>web browser</i>. 2. Sistem menampilkan visualisasi peta kerjasama peneliti untuk semua fakultas.



Gambar 3. 4 Diagram Aktifitas Kasus Kebutuhan UC-P01

3.1.4.2. Deskripsi Kasus Kebutuhan UC-P02

Kasus kebutuhan kode UC-P02 merupakan kasus kebutuhan melihat peta kerjasama peneliti setiap fakultas. Rincian alur kasus melihat peta kerjasama peneliti setiap fakultas dijelaskan pada Tabel 3. 3 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 5.



Gambar 3. 5 Diagram Aktifitas Kasus Kebutuhan UC-P02

Tabel 3. 3 Rincian Alur Kasus Penggunaan UC-P02

Nama Case	Use	Melihat peta kerjasama peneliti setiap fakultas
Nomor		UC-P02
Aktor		Pengguna
Kondisi Awal		Graf masih menampilkan peta kerjasama peneliti semua fakultas
Kondisi akhir		Graf menampilkan peta kerjasama peneliti sesuai fakultas yang dipilih
Alur Normal		1. Pengguna memilih tab <i>nodes</i> .

	<ol style="list-style-type: none"> 2. Sistem menampilkan <i>pie charts</i> pada tab <i>nodes</i>. 3. Pengguna memilih fakultas tertentu pada <i>pie charts</i>. 4. Sistem menampilkan visualisasi graf peta kerjasama peneliti sesuai fakultas yang dipilih.
--	---

3.1.4.3. Deskripsi Kasus Kebutuhan UC-P03

Kasus kebutuhan kode UC-P03 merupakan kasus kebutuhan melihat persentase persebaran peneliti setiap fakultas. Rincian alur kasus melihat persentase persebaran peneliti setiap fakultas dijelaskan pada Tabel 3. 4 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 6.



Gambar 3. 6 Diagram Aktifitas Kasus Kebutuhan UC-P03

Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-P03

Nama Use Case	Melihat persentase persebaran peneliti setiap fakultas
Nomor	UC-P03
Aktor	Pengguna
Kondisi Awal	Persentase persebaran peneliti belum terlihat
Kondisi akhir	Persentase persebaran peneliti terlihat
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih tab <i>nodes</i>. 2. Pengguna mengarahkan pilihan ke fakultas tertentu. 3. Sistem menampilkan persentase persebaran peneliti untuk fakultas yang dipilih.

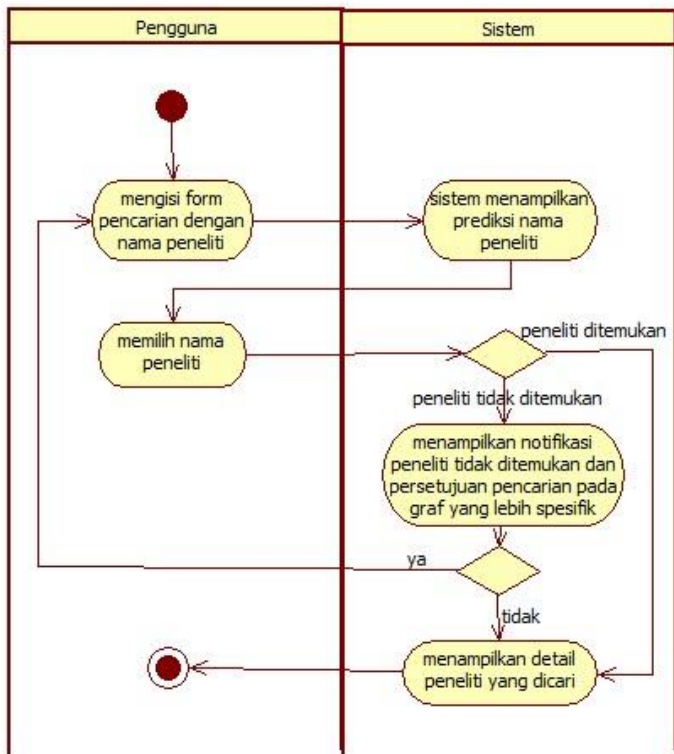
3.1.4.4. Deskripsi Kasus Kebutuhan UC-P04

Kasus kebutuhan kode UC-P04 merupakan kasus kebutuhan melakukan pencarian peneliti. Rincian alur kasus melakukan pencarian peneliti dijelaskan pada Tabel 3. 5 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 7.

Tabel 3. 5 Rincian Alur Kasus Penggunaan UC-P04

Nama Use Case	Melakukan pencarian peneliti
Nomor	UC-P04
Aktor	Pengguna
Kondisi Awal	Form pencarian masih kosong
Kondisi akhir	Sistem menampilkan peneliti yang dicari
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna mengisi nama peneliti yang dicari. 2. Sistem menampilkan prediksi nama peneliti yang dicari.

	<p>3. Pengguna memilih nama peneliti yang dicari.</p> <p>4. Sistem menampilkan detail peneliti yang dicari.</p> <p>A1. Sistem tidak menemukan peneliti yang dicari pada graf yang sedang aktif.</p>
	<p>A1.1 Sistem menampilkan notifikasi bahwa peneliti yang dicari tidak ditemukan pada graf yang sedang aktif dan memberikan pilihan apakah pengguna ingin menemukan peneliti pada graf yang lebih spesifik.</p> <p>A1.2 Pengguna memilih menampilkan peneliti yang dicari pada graf yang lebih spesifik.</p> <p>A2. Pengguna tidak memilih menampilkan peneliti yang dicari pada graf yang lebih spesifik.</p> <p>A1.3 Kembali ke alur 4</p> <p>A2.1 Kembali ke alur 1.</p>



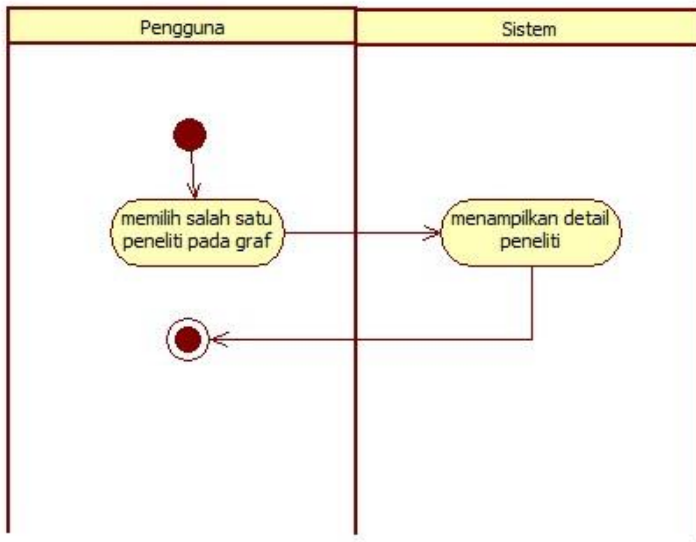
Gambar 3. 7 Diagram Aktifitas Kasus Kebutuhan UC-P04

3.1.4.5. Deskripsi Kasus Kebutuhan UC-P05

Kasus kebutuhan kode UC-P05 merupakan kasus kebutuhan melihat detail peneliti. Rincian alur kasus melihat detail peneliti dijelaskan pada Tabel 3. 6 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 8.

Tabel 3. 6 Rincian Alur Kasus Penggunaan UC-P05

Nama Case	Melihat detail peneliti
Nomor	UC-P05
Aktor	Pengguna
Kondisi Awal	Pengguna telah masuk ke sistem
Kondisi akhir	Detail pengguna tampil
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih salah satu peneliti pada graf. 2. Sistem menampilkan detail peneliti.



Gambar 3. 8 Diagram Aktifitas Kasus Kebutuhan UC-P05

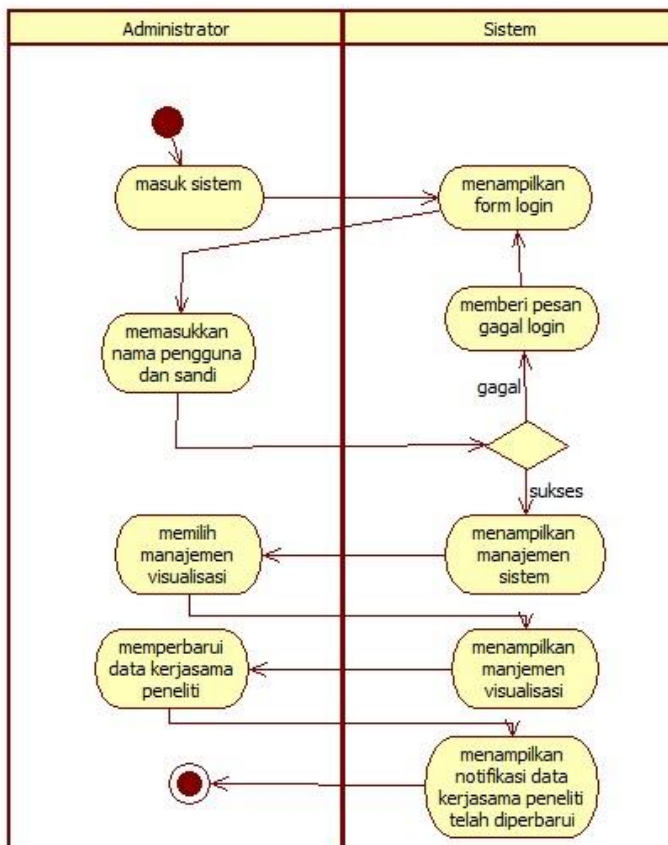
3.1.4.6. Deskripsi Kasus Kebutuhan UC-P06

Kasus kebutuhan kode UC-P06 merupakan kasus kebutuhan memperbarui data peta kerjasama peneliti. Rincian alur kasus

memperbarui data peta kerjasama peneliti dijelaskan pada Tabel 3. 7 dan diagram aktifitas kasus dijelaskan pada Gambar 3. 9.

Tabel 3. 7 Rincian Alur Kasus Penggunaan UC-P06

Nama Case	Memperbarui data peta kerjasama peneliti
Nomor	UC-P06
Aktor	Administrator
Kondisi Awal	Data kerjasama peneliti belum diperbarui
Kondisi akhir	Data kerjasama peneliti telah diperbarui
Alur Normal	<ol style="list-style-type: none"> 1. Administrator masuk ke dalam sistem. 2. Sistem menampilkan halaman login. 3. Administrator memasukkan nama pengguna dan sandi. A1. Nama pengguna dan sandi salah 4. Sistem menampilkan manajemen sistem. 5. Administrator memilih tab manajemen visualisasi. 6. Sistem menampilkan manajemen visualisasi. 7. Administrator melakukan pembaruan data kerjasama peneliti. 8. Sistem menampilkan notifikasi pembaruan data kerjasama peneliti berhasil.
	A1. Sistem memberi pesan gagal login. A1.1. Kembali ke alur 3.



Gambar 3. 9 Diagram Aktifitas Kasus Kebutuhan UC-P06

3.2 Perancangan

Sub bab berikut membahas tentang perancangan dari kaskas visualisasi kerjasama peneliti. Sub bab berikut akan membahas rancangan antarmuka, rancangan proses dan rancangan basis data.

3.2.1 Perancangan Basis Data

Pada sub bab ini akan dijelaskan bagaimana rancangan basis data yang digunakan pada kaskas visualisasi kerjasama peneliti. Basis data pada sistem yang dibangun pada tugas akhir ini menggunakan sistem manajemen basis data SQL Server. SQL Server digunakan untuk menyimpan data tugas akhir, data peneliti, data relasi antar peneliti, daftar *term* unik untuk semua dokumen beserta pembobotannya, daftar semua *term* untuk setiap peneliti, dan data perhitungan algoritma. *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) dari basis data sistem ini dapat dilihat pada Lampiran C tentang *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM).

3.2.1.1. Rancangan Tabel Kata Dasar (tran_bobot_katadasar)

Tabel tran_bobot_katadasar digunakan untuk menyimpan data kata dasar bahasa indonesia. Kata dasar digunakan pada tahap *stemming* menggunakan algoritma Porter Bahasa Indonesia. Detil atribut tabel tran_bobot_katadasar dijelaskan pada Tabel 3. 8.

Tabel 3. 8 Atribut Tabel tran_bobot_katadasar

Nama Kolom	Tipe Data	Keterangan
id_katadasar	Integer	<i>Primary Key</i> dari table tran_bobot_katadasar
katadasar	Varchar (50)	Kata dasar

tipe_katadasar	Varchar (20)	Jenis kata dasar (kata benda, kata sifat, kata kerja)
----------------	--------------	---

3.2.1.2. Rancangan Tabel Kata Indeks (tran_bobot_kata_indeks)

Tabel tran_bobot_kata_indeks digunakan untuk menyimpan daftar *term* beserta pembobotannya. Daftar *term* tersebut merupakan daftar *term* yang sudah unik berasal dari *file* indeks yang sudah di *generate* menggunakan kerangka kerja Zend Search Lucene. Detil atribut tabel tran_bobot_kata_indeks dijelaskan pada Tabel 3. 9.

Tabel 3. 9 Atribut Tabel tran_bobot_kata_indeks

Nama Kolom	Tipe Data	Keterangan
kata	Varchar(30)	<i>term</i>
df	Integer	Frekuensi dokumen yang mengandung <i>term</i> tersebut
idf_norm	Float	Bentuk normal dari frekuensi dokumen

3.2.1.3. Rancangan Tabel Kata Dokumen (tran_bobot_kata_dokumen)

Tabel tran_bobot_kata_dokumen digunakan untuk menyimpan daftar *term* beserta perhitungan bobot TF-IDF setiap *term* pada setiap dokumen. Detil atribut tabel komentar dijelaskan pada Tabel 3. 10.

Tabel 3. 10 Atribut Tabel tran_bobot_kata_dokumen

Nama Kolom	Tipe Data	Keterangan
id_dokumen	Varchar (50)	Nomor Induk Mahasiswa (NIM) sebagai ID dokumen tugas akhir
kata	Varchar (50)	<i>term</i>
tf	Integer	Frekuensi munculnya <i>term</i> pada dokumen
tf_norm	Float	Bentuk normal dari frekuensi munculnya <i>term</i> pada dokumen
tf_idf_norm	Float	Bobot TF-IDF <i>term</i> pada dokumen

3.2.1.4. Rancangan Tabel Vektor Topik (tran_bobot_vektor_topik)

Tabel tran_bobot_vektor_topik digunakan untuk menyimpan daftar *term* untuk setiap peneliti beserta pembobotan TF-IDF masing-masing *term*. Pada tabel tran_bobot_kata_dokumen daftar *term* disimpan dengan ID mahasiswa. Sedangkan pada tabel tran_bobot_vektor topik ini setiap *term* disimpan menggunakan ID peneliti yang membimbing tugas akhir mahasiswa tersebut. Detil atribut tabel tran_bobot_vektor_topik dapat dilihat pada Tabel 3. 11.

Tabel 3. 11 Atribut Tabel tran_bobot_vektor_topik

Nama Kolom	Tipe Data	Keterangan
nidn	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID dokumen tugas akhir
kata	Varchar (30)	<i>term</i>
tf_idf_norm	Float	Bobot TF-IDF <i>term</i> pada dokumen

3.2.1.5. Rancangan Tabel Vektor 1 (tran_bobot_vektor_1)

Tabel tran_bobot_vektor_1 digunakan untuk menyimpan daftar *term* yang unik setiap peneliti beserta rata-rata TF-IDF *term* jika terdapat *term* yang sama pada dokumen berbeda. Dari daftar *term* setiap peneliti diambil beberapa *term* dengan bobot TF-IDF teratas, yang mana daftar *term* teratas inilah yang menjadi representasi topik penelitian setiap peneliti. Detil atribut tabel tran_bobot_vektor_1 dapat dilihat pada Tabel 3. 12.

Tabel 3. 12 Atribut Tabel tran_bobot_vektor_1

Nama Kolom	Tipe Data	Keterangan
nidn	Nchar (10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID dokumen tugas akhir
kata	Varchar (50)	<i>term</i>

tf_idf_norm	Float	Bobot TF-IDF <i>term</i> pada dokumen
-------------	-------	---

3.2.1.6. Rancangan Tabel Vektor 2
(tran_bobot_vektor_2)

Tabel tran_bobot_vektor_2 digunakan untuk menyimpan bobot topik antar peneliti. Bobot topik merupakan hasil perhitungan dari daftar *term* kedua peneliti yang merepresentasikan kemiripan topik keduanya. Detil atribut tabel tran_bobot_vektor_2 dapat dilihat di Tabel 3. 13.

Tabel 3. 13 Atribut Tabel tran_bobot_vektor_2

Nama Kolom	Tipe Data	Keterangan
nidn1	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID Peneliti 1
nidn2	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID Peneliti 2
kali_tambah	Float	
kuadrat_akar_nidn1	Float	
kuadrat_akar_nidn2	Float	
final	Float	Bobot kemiripan topik kedua peneliti

3.2.1.7. Rancangan Tabel Dosen Single (tran_bobot_dosen_single)

Tabel tran_bobot_dosen_single digunakan untuk menyimpan detail setiap peneliti. Detil atribut tabel file dapat dilihat pada Tabel 3. 14.

Tabel 3. 14 Atribut Tabel tran_bobot_dosen_single

Nama Kolom	Tipe Data	Keterangan
nidn	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID Peneliti
jumlah_dokumen	Integer	Jumlah dokumen tugas akhir yang dibimbing peneliti
id_node	Integer	
nip_baru	Varchar(50)	Nomor Induk Pegawai (NIP) baru peneliti
jumlah_kemiripan	Integer	Jumlah peneliti yang lain yang mirip

3.2.1.8. Rancangan Tabel Dosen Peneliti (tran_bobot_dosen_peneliti)

Tabel tran_bobot_dosen_peneliti digunakan untuk menyimpan bobot hubungan kerjasama antar peneliti. Data yang disimpan meliputi bobot kerjasama penelitian yang dikerjakan secara bersama-sama dan bobot berdasarkan kemiripan topik. Hasil akhir bobot kerjasama peneliti merupakan hasil perkalian dari bobot kerjasama penelitian bersama dengan bobot kemiripan topik.

Detil atribut tabel tran_bobot_dosen_peneliti dapat dilihat pada Tabel 3. 15.

Tabel 3. 15 Atribut Tabel tran_bobot_dosen_peneliti

Nama Kolom	Tipe Data	Keterangan
nidn1	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID Peneliti 1
nidn2	Nchar(10)	Nomor Induk Dosen Nasional (NIDN) sebagai ID Peneliti 2
bobot_peneliti	Float	Bobot kerjasama penelitian/pembimbing tugas akhir bersama
bobot_topik	Float	Bobot kemiripan topik
final	Float	Hasil akhir bobot kerjasama peneliti

3.2.1.9. Rancangan Tabel Admin (tref_user_admin)

Tabel tref_user_admin digunakan untuk menyimpan nama pengguna dan sandi administrator. Detil atribut tabel tref_user_admin dapat dilihat pada Tabel 3. 16.

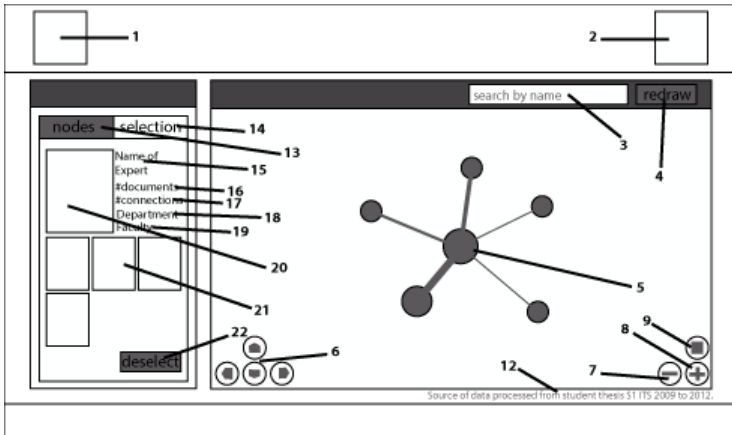
Tabel 3. 16 Atribut Tabel tref_user_admin

Nama Kolom	Tipe Data	Keterangan
username	Varchar(50)	Nama pengguna administrator
password	Varchar(50)	Sandi administrator menggunakan enkripsi MD5

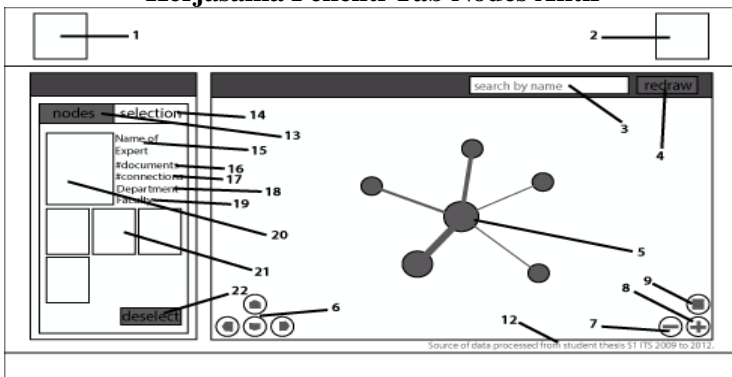
3.2.2 Perancangan Antarmuka

Pada sub bab ini akan dibahas secara mendetail dari rancangan antarmuka kakas visualisasi.

3.2.2.1 Antarmuka Halaman Visualisasi Peta Kerjasama Peneliti



Gambar 3. 10 Rancangan Antar Muka Visualisasi Peta Kerjasama Peneliti Tab Nodes Aktif



Gambar 3. 11 Rancangan Antar Muka Visualisasi Peta Kerjasama Peneliti Tab Selection Aktif

Gambar 3. 10 dan Gambar 3. 11 merupakan gambar rancangan visualisasi peta kerjasama peneliti. Gambar 3. 10 merupakan tampilan ketika kakas visualisasi dibuka, tab yang aktif adalah tab *nodes*. Gambar 3. 11 merupakan tampilan ketika salah satu peneliti dipilih, tab yang aktif adalah tab *selection*. Berikut penjelasan masing–masing nomor yang tertera pada Gambar 3. 10 dan Gambar 3. 11.

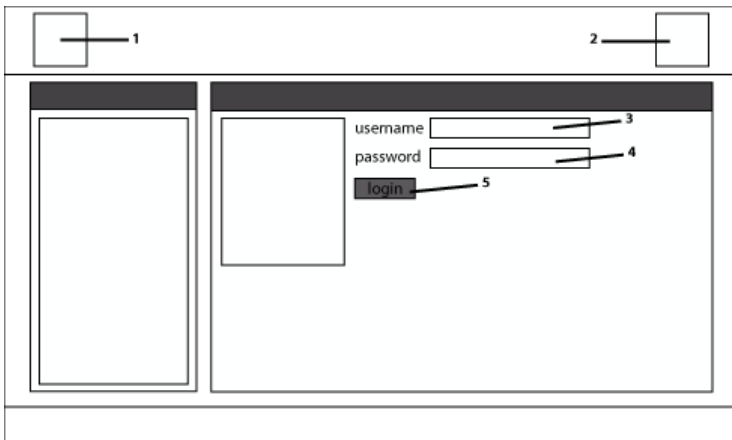
1. Logo Sistem Repositori Peneliti / Resources ITS (RESITS).
2. Logo ITS.
3. Berupa *textbox* untuk melakukan pencarian peneliti. Ketika pengguna mengetikkan nama peneliti, maka akan muncul *dropdown* berisi prediksi nama peneliti yang dicari.
4. Berupa tombol *redraw* berguna untuk menggambar ulang visualisasi graf peta kerjasama peneliti seperti pertama kali dibuka.
5. Berupa graf visualisasi peta kerjasama peneliti. Setiap lingkaran (*node*) merepresentasikan peneliti. Semakin besar lingkaran maka semakin banyak tugas akhir yang dibimbing. Warna lingkaran merepresentasikan area penelitian (fakultas) peneliti. Hubungan kerjasama berdasarkan kemiripan topik penelitian direpresentasikan oleh garis (*edge*). Semakin dekat dan tebal garis, maka semakin dekat kerjasama berdasarkan kemiripan topik penelitian antar peneliti.
6. Berupa tombol navigasi visualisasi, ke atas, ke bawah, ke kanan dan ke kiri. Untuk keperluan navigasi bisa juga menggunakan *keyboard*.
7. Berupa tombol, berfungsi untuk memperkecil visualisasi.
8. Berupa tombol, berfungsi untuk memperbesar visualisasi.
9. Berupa tombol, berfungsi untuk mengembalikan visualisasi seperti semula.

10. Berupa *pie charts*, berfungsi menampilkan persentase persebaran peneliti setiap fakultas, yaitu dengan mengarahkan *mouse pointer* ke salah satu fakultas. Kemudian ketika diklik salah satu fakultas maka visualisasi graf akan menampilkan peta kerjasama peneliti berdasarkan fakultas yang dipilih.
11. Berupa daftar nama fakultas, berfungsi untuk menunjukkan keberadaan peneliti dengan fakultas tertentu pada visualisasi graf, yaitu dengan mengarahkan *mouse pointer* pada nama fakultas tertentu.
12. Berupa keterangan sumber data yang diproses untuk keperluan visualisasi peta kerjasama peneliti.
13. Berupa tab dengan label *nodes*, berisi area penelitian (*Research Area*) dan persentase persebaran peneliti (*Distributions*).
14. Berupa tab dengan label *selection*, berisi detail informasi ketika salah satu peneliti dipilih.
15. Berupa label nama peneliti.
16. Berupa label jumlah dokumen tugas akhir yang dibimbing oleh peneliti.
17. Berupa label jumlah peneliti lain yang mirip dengan peneliti yang dipilih, jumlah sesuai dengan visualisasi graf yang sedang aktif.
18. Berupa label menunjukkan jurusan peneliti.
19. Berupa label menunjukkan fakultas peneliti.
20. Berupa foto peneliti.
21. Berupa daftar foto peneliti-peneliti lain yang memiliki kemiripan kerjasama berdasarkan topik penelitian dengan peneliti yang dipilih. Kemiripan diurutkan berdasarkan urutan daftar foto peneliti.

3.2.2.2 Antarmuka Halaman Login Manajemen Sistem Repositori Peneliti

Gambar 3. 12 merupakan rancangan halaman login manajemen Sistem Repositori Peneliti. Halaman tersebut

digunakan administrator untuk masuk ke manajemen Sistem Repositori Peneliti. Di dalam manajemen tersebut terdapat tiga macam manajemen, yaitu manajemen peneliti, manajemen dokumen penelitian dan manajemen visualisasi. Dalam tugas akhir ini hanya akan dibangun manajemen visualisasi untuk keperluan visualisasi peta kerjasama peneliti.



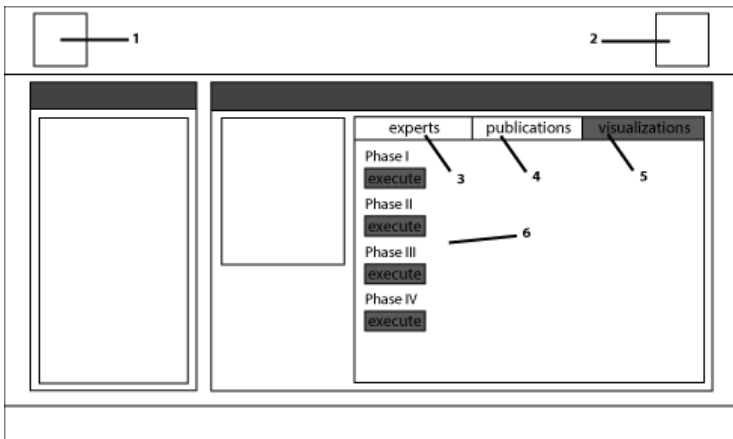
**Gambar 3. 12 Rancangan Antarmuka Halaman Login
Manajemn Sistem Repositori Peneliti**

Gambar 3. 12 merupakan rancangan antarmuka halaman login Sistem Repositori Peneliti. Berikut penjelasan untuk masing-masing nomor yang tertera pada Gambar 3. 12.

1. Logo Sistem Repositori Peneliti / Resources ITS (RESITS).
2. Logo ITS.
3. Berupa *textboxt* untuk memasukkan nama pengguna administrator.
4. Berupa *textboxt* untuk memasukkan sandi administrator.
5. Berupa tombol untuk masuk sistem manajemen.

3.2.2.3 Antarmuka Halaman Manajemen Visualisasi Sistem Repositori Peneliti

Gambar 3. 13 merupakan rancangan halaman manajemen visualisasi Sistem Repositori Peneliti. Halaman ini diakses administrator apabila ingin memperbarui data visualisasi peta kerjasama peneliti.



Gambar 3. 13 Rancangan Antarmuka Halaman Manajemen Sistem Repositori Peneliti Tab Visualisasi Aktif

Gambar 3. 13 merupakan gambar rancangan antarmuka halaman manajemen visualisasi Sistem Repositori Peneliti, yaitu pada tab manajemen visualisasi. Berikut penjelasan untuk masing-masing nomor yang tertera pada Gambar 3. 13.

1. Logo Sistem Repositori Peneliti / Resources ITS (RESITS).
2. Logo ITS.
3. Berupa tab manajemen pengindeksan data peneliti.
4. Berupa tab manajemen pengindeksan data penelitian.
5. Berupa tab manajemen visualisasi.

6. Berupa daftar tahapan-tahapan proses perbaruan data visualisasi peta kerjasama peneliti.

3.2.3 Perancangan Proses Aplikasi









Pada subbab ini akan dibahas secara mendetil dari rancangan proses kakas visualisasi peta kerjasama peneliti berdasarkan topik penelitian untuk memenuhi kebutuhan fungsionalnya.

3.2.3.1 Proses Pengindeksan Data Penelitian Tugas Akhir

Pengindeksan pada tahap ini digunakan untuk mendapatkan semua kata (*term*) yang unik untuk seluruh dokumen tugas akhir. Dalam pengindeksan ini dilakukan menggunakan kerangka kerja Zend Search Lucene. Pengindeksan merupakan pembuatan *file binary* semacam kamus data, yaitu tempat dimana dokumen-dokumen diekstrak informasinya (misal : judul, isi dokumen) kemudian diurutkan menjadi satu kesatuan yang terindeks. Untuk keperluan visualisasi kerjasama berdasarkan topik ini informasi yang diekstrak berasal dari dokumen tugas akhir, yaitu judul, abstraksi, kata kunci, id mahasiswa, dan id peneliti-peneliti yang membimbing untuk setiap penelitian tugas akhir. Proses pengindeksan dapat dilihat pada Gambar 3. 14, dan contoh *file* indeks hasil pengindeksan dapat dilihat pada Gambar 3. 15.



Gambar 3. 14 Proses Pengindeksan Data Penelitian Tugas Akhir

 _9a.cfs	6/4/2014 11:47 PM	CFS File	3 KB
 _9b.cfs	6/4/2014 11:47 PM	CFS File	911 KB
 _32.cfs	6/4/2014 11:44 PM	CFS File	2 KB
 _33.cfs	6/4/2014 11:44 PM	CFS File	891 KB
 _66.cfs	6/4/2014 11:45 PM	CFS File	2 KB
 _67.cfs	6/4/2014 11:45 PM	CFS File	855 KB
 _ce.cfs	6/4/2014 11:48 PM	CFS File	2 KB
 _cf.cfs	6/4/2014 11:48 PM	CFS File	896 KB

Gambar 3. 15 Contoh File Indeks Hasil Pengindeksan

3.2.3.2 Proses Tokenisasi dan Menghilangkan Stopwords

Dari *file* indeks pada tahap sebelumnya, maka selanjutnya dilakukan tokenisasi untuk mendapatkan semua daftar *term* yang unik. Setelah mendapatkan daftar semua *term* yang unik untuk semua penelitian tugas akhir, tahap selanjutnya adalah menghilangkan *term* yang termasuk *stopwords*. *Stopwords* adalah kata umum (*common words*) yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna (misal: “yang”, “di”, “ke”). Proses ini dapat dilihat pada Gambar 3. 16.

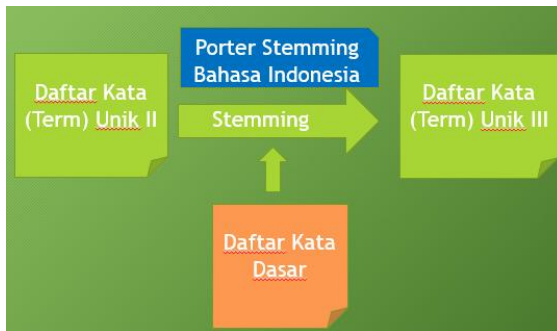


Gambar 3. 16 Proses Tokenisasi dan Menghilangkan Stopwords

3.2.3.3 Proses Melakukan Stemming

Tahap selanjutnya adalah melakukan stemming. Stemming merupakan suatu proses yang menstransformasikan kata-kata yang terdapat dalam suatu dokumen kata dasar (*root word*) dengan

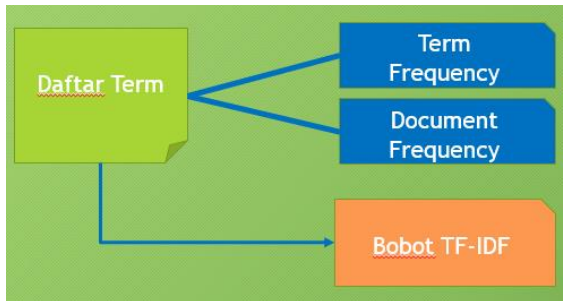
menggunakan aturan-aturan tertentu. Algoritma *stemming* yang digunakan dalam tugas akhir ini adalah Algoritma *Porter Stemming* untuk Bahasa Indonesia. Proses ini dapat dilihat pada Gambar 3. 17.



Gambar 3. 17 Proses Melakukan Stemming

3.2.3.4 Proses Pembobotan Kata

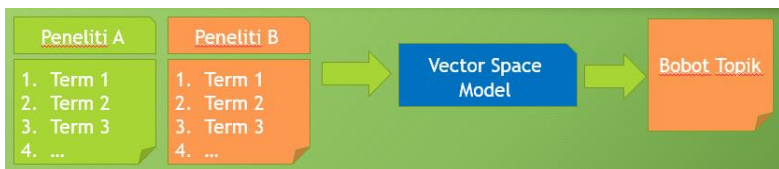
Tahap selanjutnya yaitu melakukan pembobotan untuk setiap *term*. Pembobotan ini menggunakan algoritma TF-IDF, meliputi *term frequency* dan *inverse document frequency*. Seperti yang sudah dijelaskan pada bab sebelumnya, pembobotan ini digunakan untuk mengetahui hubungan *term* dengan dokumen. Dari pembobotan ini akan didapatkan daftar *term* (kata) penting untuk setiap peneliti. Proses ini dapat dilihat pada Gambar 3. 18.



Gambar 3. 18 Proses Pembobotan Kata

3.2.3.5 Proses Penghitungan Bobot Topik Antar Peneliti

Bobot relasi topik peneliti merupakan besarnya hubungan antar peneliti berdasarkan kemiripan topiknya. Kemiripan topik ini didapatkan dari banyaknya *term* yang dipakai bersama oleh masing-masing peneliti. Kemiripan topik antar peneliti dihitung menggunakan algoritma *Vector Space Model* yang sudah dijelaskan pada bab sebelumnya. Proses ini dapat dilihat pada Gambar 3. 19.

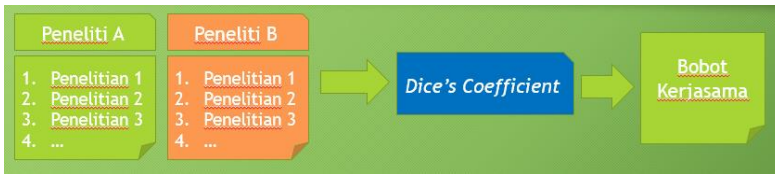


Gambar 3. 19 Proses Penghitungan Bobot Topik Antar Peneliti

3.2.3.6 Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti

Selain bobot relasi topik antar peneliti, kerjasama antara peneliti berdasarkan topik ini juga dipengaruhi oleh bobot kerjasama penelitian tugas akhir yang dilakukan secara bersama.

Relasi kerjasama peneliti merupakan hubungan antar peneliti berdasarkan seringnya peneliti bekerjasama dengan peneliti lain. Relasi kerjasama ini dihitung menggunakan algoritma *Dice's Coefficient* yang sudah dijelaskan pada bab sebelumnya. Proses ini dapat dilihat pada Gambar 3. 20.



Gambar 3. 20 Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti

3.2.3.7 Proses Penghitungan Bobot Kerjasama Peneliti Berdasarkan Topik Penelitian

Besarnya bobot kerjasama berdasarkan topik penelitian antar peneliti dipengaruhi oleh besarnya bobot relasi topik dan bobot kerjasama penelitian antara kedua peneliti. Bobot kerjasama berdasarkan topik penelitian antar peneliti merupakan hasil perkalian dari besarnya bobot relasi topik dan besarnya bobot relasi kerjasama penelitian antar peneliti. Proses ini dapat dilihat pada Gambar 3. 21.



Gambar 3. 21 Proses Penghitungan Bobot Kerjasama Peneliti Berdasarkan Topik Penelitian

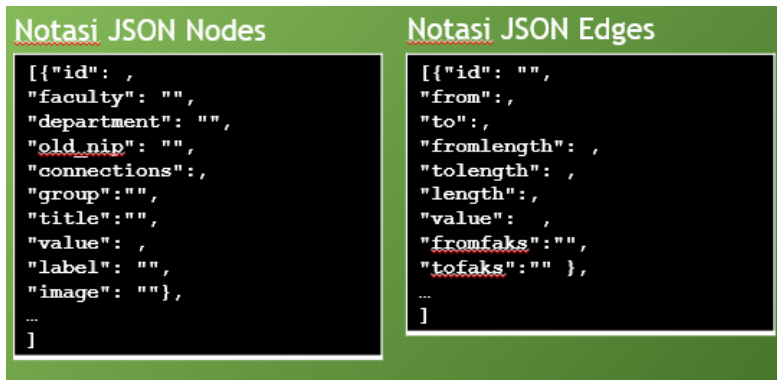
3.2.3.8 Proses Pemodelan Graf Dalam Bentuk JSON

Pada tahap ini hubungan antar peneliti dimodelkan dalam bentuk graf dengan notasi JSON. Dari notasi-notasi yang sudah terbentuk, selanjutnya akan di-generate file JSON. File JSON inilah yang akan diakses untuk divisualisasikan dalam bentuk visualisasi berbasis web. Proses ini dapat dilihat pada Gambar 3. 22.

Notasi yang terbentuk merupakan representasi dari kerjasama peneliti menggunakan pemodelan graf. *Node* pada graf merepresentasikan setiap peneliti. Bobot antar *node* direpresentasikan oleh *edge* (garis) yang menghubungkan *node* peneliti. Dalam visualisasi ini dibutuhkan dua bentuk file JSON, yaitu notasi *node* yang menyimpan informasi setiap peneliti dan notasi *edge* yang menyimpan bobot hubungan kerjasama peneliti berdasarkan topik. Notasi *node* setiap peneliti menyimpan beberapa informasi seperti nama, jumlah dokumen tugas akhir, jumlah kemiripan dengan dosen lain, jurusan, fakultas peneliti. Notasi yang terbentuk secara mendetail dapat dilihat pada Gambar 3. 23.



Gambar 3. 22 Proses Pemodelan Graf Dalam Bentuk JSON



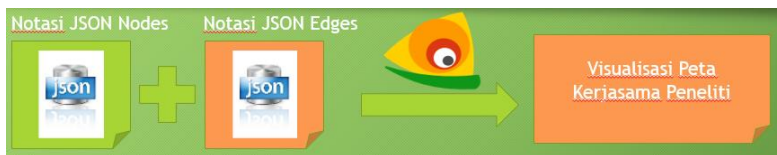
Gambar 3. 23 Notasi JSON

3.2.3.9 Proses Visualisasi Graf Berbasis Web

Pada subbab ini akan dibahas secara mendetil dari proses visualisasi peta kerjasama peneliti berbasis web.

3.2.3.9.1 Proses Inisiasi Visualisasi Graf

Pada tahap inisiasi visualisasi graf ini peta kerjasama peneliti akan divisualisasikan menggunakan pemodelan graf. Dalam tugas akhir ini diperlukan kerangka kerja yang mampu memvisualisasikan pemodelan graf berbasis web, yaitu Vis.Js. Notasi graf yang sudah terbentuk pada tahap sebelumnya diakses untuk divisualisasikan menggunakan kerangka kerja Vis.Js. Proses ini dapat dilihat pada Gambar 3. 24.



Gambar 3. 24 Proses Inisiasi Visualisasi Graf

3.2.3.9.2 Proses Menampilkan Detail Peneliti dan Mengurutkan Peneliti-Peneliti yang Mirip Topik Penelitiannya

Pada tahap visualisasi graf ini juga menangani proses menampilkan detail informasi peneliti ketika salah satu *node* graf dari visualisasi dipilih atau dengan pencarian peneliti. Detail informasi yang ditampilkan meliputi nama peneliti, jumlah penelitian tugas akhir yang pernah dibimbing, jumlah kemiripan topik penelitian dengan peneliti lain pada graf yang sedang aktif, jurusan serta fakultas.

Selain itu detail peneliti juga akan menampilkan peneliti-peneliti yang mirip topik penelitiannya. Peneliti-peneliti yang mirip ini diurutkan berdasarkan tingkat kemiripannya. Data peneliti-peneliti yang mirip ini diambil dari graf yang visualisasi sedang aktif. Data tersebut kemudian diproses untuk mendapatkan peneliti-peneliti berdasarkan urutan tingkat kemiripan topik penelitiannya.

3.2.3.9.3 Proses Menghitung dan Menampilkan Persentase Persebaran Peneliti

Pada visualisasi peta kerjasama peneliti ini juga ditampilkan persentase persebaran peneliti untuk setiap area peneliti (fakultas). Data persentase didapatkan dari notasi *node* JSON, data di-*filter* berdasarkan area penelitiannya, kemudian dihitung persentase banyaknya peneliti untuk setiap area peneliti.

3.2.3.9.4 Proses Pencarian Peneliti

Pada visualisasi peta kerjasama peneliti ini, pengguna juga dapat melakukan pencarian berdasarkan nama peneliti. Pengguna cukup memasukkan nama peneliti pada kolom pencarian, kemudian akan muncul prediksi dari nama peneliti yang akan dicari. Pengguna cukup memilih nama peneliti yang akan dicari,

jika peneliti ditemukan pada graf visualisasi yang sedang aktif, maka *node* peneliti pada graf akan terpilih dan muncul detail informasi peneliti. Jika peneliti yang dicari tidak ditemukan pada graf visualisasi yang sedang aktif, maka akan muncul pilihan untuk menampilkan peneliti dengan graf yang lebih spesifik. Informasi peneliti yang dicari berasal dari notasi *node* JSON, dari data semua peneliti yang tersimpan dalam notasi tersebut di-*filter* berdasarkan nama peneliti hingga mendapatkan peneliti yang dicari.

3.2.3.9.5 Proses Menampilkan Peta Kerjasama Peneliti Setiap Area Peneliti

Pada visualisasi peta kerjasama peneliti ini, pengguna juga dapat menampilkan peta kerjasama peneliti berdasarkan area penelitiannya (fakultas). Data peneliti yang tersimpan dalam notasi *node* JSON di-*filter* berdasarkan area penelitiannya. Setelah peneliti-peneliti dengan area penelitian tertentu didapatkan, selanjutnya juga dilakukan *filter* terhadap notasi *edge* yang berisi data bobot hubungan kerjasama antar *node* peneliti. Sama halnya dengan notasi *node* JSON, notasi *edge* JSON di-*filter* berdasarkan area penelitiannya.

BAB IV IMPLEMENTASI

Pada bab ini akan dibahas implementasi dari rancangan sistem yang ditulis pada Bab 3. Namun, tidak menutup kemungkinan adanya perubahan-perubahan dari rancangan tersebut apabila memang diperlukan.

4.1 Lingkungan Implementasi

Dalam merancang perangkat lunak ini digunakan beberapa perangkat pendukung sebagai berikut.

4.1.1 Lingkungan Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem adalah komputer dengan spesifikasi, AMD Athlon™ II X2 250 Processor 3.00 GHz dan 4GB memory.

4.1.2 Lingkungan Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan untuk pengembangan sistem adalah sebagai berikut:

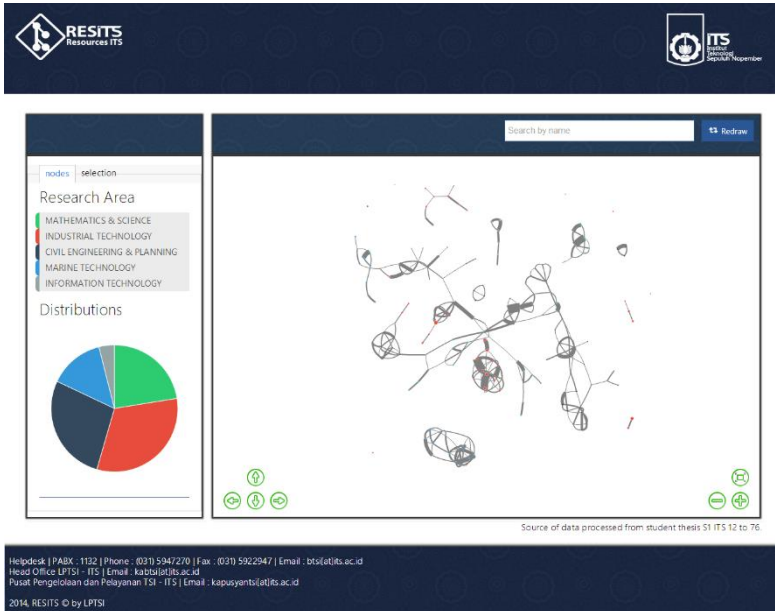
- Microsoft Windows 8.1 sebagai sistem operasi.
- Notepad++ 5.9.6.2 sebagai IDE untuk implementasi aplikasi.
- SQL Server 2005 sebagai aplikasi manajemen basis data.
- Star UML 5.0.2.1570 untuk merancang diagram kasus penggunaan.

4.2 Implementasi Antarmuka

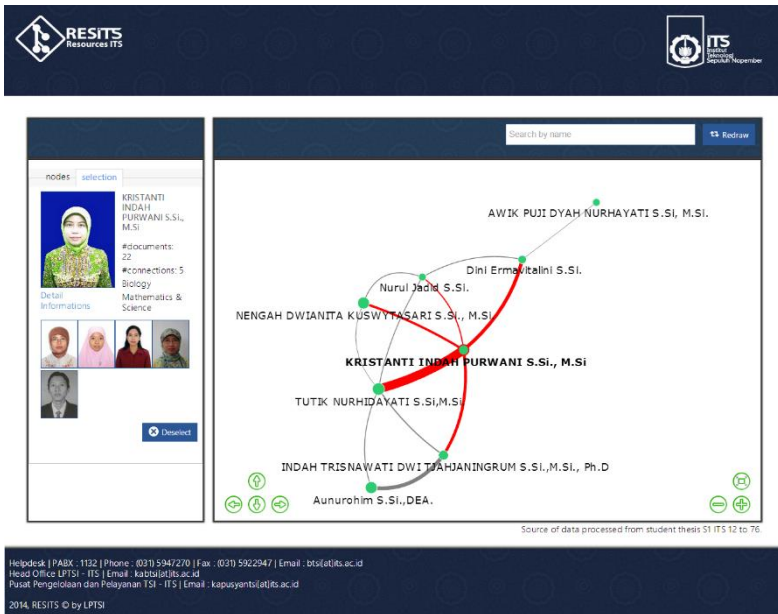
Pada subbab ini akan dibahas implementasi antarmuka berdasarkan rancangan antarmuka yang telah dibahas pada bab 3. Antarmuka yang akan dibahas terdiri dari antarmuka visualisasi peta kerjasama peneliti dan antarmuka manajemen visualisasi.

4.2.1 Antarmuka Halaman Visualisasi Peta Kerjasama Peneliti

Berikut implementasi halaman visualisasi peta kerjasama peneliti seperti pada Gambar 4. 1. Gambar 4. 1 merupakan halaman visualisasi peta kerjasama peneliti yang diakses oleh pengguna.



(a)



(b)

Gambar 4. 1 (a,b) Halaman Visualisasi Peta Kerjasama Peneliti

Pada Gambar 4. 1(a) merupakan tampilan halaman visualisasi peta kerjasama peneliti ketika pertama kali dibuka oleh pengguna. Peta kerjasama peneliti direpresentasikan oleh visualisasi graf pada sebelah kanan. Peneliti direpresentasikan oleh lingkaran, dan hubungan kerjasama peneliti berdasarkan kemiripan topik direpresentasikan oleh garis. Pada bagian tab, tab *nodes* secara otomatis akan terpilih, berisi area penelitian dan persebaran peneliti setiap fakultas. Sedangkan Gambar 4. 1(b) yaitu ketika pengguna memilih salah satu peneliti yang direpresentasikan oleh lingkaran pada visualisasi graf, maka tab *selection* otomatis akan terpilih, yang mana tab tersebut berisi detail informasi peneliti yang

dipilih, seperti jumlah dokumen tugas akhir yang dibimbing, jumlah hubungan kerjasama kemiripan topik pada graf yang sedang aktif, jurusan, fakultas, dan peneliti-peneliti yang memiliki kedekatan kemiripan kerjasama berdasarkan topik. Pada bagian kanan atas terdapat *textbox* pencarian yang berguna untuk mencari peneliti berdasarkan nama peneliti. Sedangkan pada bagian kanan bawah terdapat informasi sumber data yang diproses untuk keperluan visualisasi peta kerjasama peneliti. Kode Sumber 4. 1 diimplementasikan pada *controller* halaman visualisasi peta kerjasama peneliti.

```
public function index()
{
    $rows = $this->index_model->get_min_max();
    $date = strtotime($rows[0]['min']);
    $new_date = date('d-m-Y', $date);

    $data['min']=$new_date;

    $date = strtotime($rows[0]['max']);
    $new_date = date('d-m-Y', $date);
    $data['max']=$new_date;
    $this->load->view('visualisasi',$data);
}
```

Kode Sumber 4. 1 Kode Sumber Controller Halaman Visualisasi Peta Kerjasama Peneliti

4.2.2 Antarmuka Halaman Login Manajemen Sistem Repositori Peneliti

Berikut implementasi halaman manajemen Sistem Repositori Peneliti seperti pada Gambar 4. 2. Gambar 4. 2 merupakan halaman login manajemen Sistem Repositori Peneliti. Administrator diminta memasukkan nama pengguna dan sandi untuk dapat masuk ke manajemen sistem.



Gambar 4. 2 Halaman Login Manajemen Sistem Repositori Peneliti

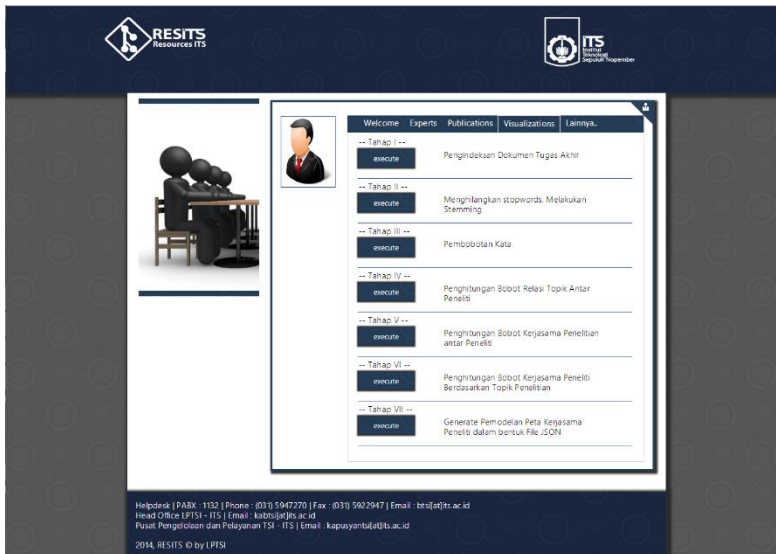
Apabila nama pengguna dan sandi yang dimasukkan administrator tidak sesuai maka muncul peringatan gagal login ke manajemen sistem. merupakan kode implementasi fungsi login. Kode Sumber 4. 2 diimplementasikan pada *controller* halaman login.

```
function index()
{
    if($this->session->userdata('logged_in'))
    {
        redirect('index.php/home',
'refresh');
    }
    else
    {
        $this->load->helper(array('form'));
        $this->load->view('home_view');
    }
}
```

Kode Sumber 4. 2 Kode Sumber Controller Halaman Login Manajemen Sistem Repositori Peneliti

4.2.3 Antarmuka Halaman Manajemen Visualisasi Sistem Repositori Peneliti

Berikut implementasi halaman manajemen Sistem Repositori Peneliti seperti pada Gambar 4. 3. Gambar 4. 3 merupakan halaman manajemen visualisasi peta kerjasama peneliti. Halaman ini digunakan administrator untuk memperbarui data peta kerjasama peneliti yang ditampilkan pada halaman visualisasi peta kerjasama peneliti. Ada beberapa tahapan untuk memperbarui data peta kerjasama peneliti sesuai proses aplikasi yang sudah dijelaskan pada bab sebelumnya.



Gambar 4. 3 Halaman Manajemen Visualisasi Sistem Repositori Peneliti

Kode Sumber 4. 3 diimplementasikan pada *controller* halaman manajemen Sistem Repositori Peneliti. Sedangkan

```
function index()
{
    if($this->session->userdata('logged_in'))
    {
        if($this->session->userdata('experts'))
        {
            $stexp=$this->session->userdata('experts');
            if($stexp==true) $data['stexp'] = "Index
updated";
            else
                $data['stexp'] = "Failed to update index";
        }
        else
            $data['stexp']=" - ";

        if($this->session->userdata('publications'))
        {
            $stpub=$this->session-
>userdata('publications');
            if($stpub==true) $data['stpub'] = "Index
updated";
            else
                $data['stpub'] = "Failed to update index";
        }
        else
            $data['stpub']=" - ";

        $session_data = $this->session-
>userdata('logged_in');
        $data['username'] = $session_data['username'];

        $expert=$this->data_model-
>get_count("experts");
        $data['exphistories']=$this->data_model-
>get_histories("experts");
        $data['countexpertsdb']=$this->index_model-
>get_num_rows("tmst_dosen");
        $data['countexperts']=$expert->count;
        $data['lastexperts']=$expert->date;
```

```

        $expert=$this->data_model-
>get_count("publications");
        $data['pubshistories']=$this->data_model-
>get_histories("publications");
        $data['countpubsdb']=$this->index_model-
>get_num_rows("tran publikasi dosen tetap");
        $data['countpubs']=$expert->count;
        $data['lastpubs']=$expert->date;

        $this->load->view('home_view', $data);
    }
    else
    {
        //If no session, redirect to login page
        redirect('index.php/login', 'refresh');
    }
}

function logout()
{
    $this->session->unset_userdata('logged_in');
    $this->session->unset_userdata('experts');
    $this->session->unset_userdata('publications');
    session_destroy();
    redirect('index.php/home', 'refresh');
}

```

Kode Sumber 4. 3 Kode Sumber Halaman Manajemen Visualisasi Sistem Repositori Peneliti

4.3 Implementasi Konsep MVC

Kakas visualisasi peta kerjasama peneliti dibangun menggunakan kerangka kerja CodeIgniter yang memiliki konsep MVC. Implementasi konsep MVC yang digunakan pada aplikasi berbasis web beberapa akan dijelaskan Pada subbab ini.

4.3.1 Implementasi Model

Kelas Model merupakan kelas yang berisi logika untuk melakukan penambahan, pengambilan, penghapusan objek ke

dalam kamus data. Kode Sumber 4. 4 merupakan kode implementasi kelas Index_model.

```
<?php
class Index_model extends CI_Model {
    function Index_model() {
        parent::__construct();
    }
    ...
}
```

Kode Sumber 4. 4 Kode Sumber Kelas Model Index_model

Kelas Model pada aplikasi berbasis web berfungsi sebagai media komunikasi data yang ditampilkan pada kelas View dan melalui kelas Controller. Kelas ini menangani penambahan, pengambilan dan penghapusan data dari kamus data. Fungsi pada kelas model dapat mengakses langsung kamus data.

4.3.2 Implementasi View

Kelas View merupakan kelas yang dapat diakses langsung oleh pengguna. Berupa kelas yang berfungsi untuk menampilkan segala informasi dari perangkat lunak kepada pengguna. Implementasi pada kelas View kakas visualisasi peta kerjasama peneliti Sistem Repositori Peneliti menggunakan HTML, JavaScript dan CSS. Kode Sumber 4. 5 merupakan implementasi View untuk tema header halaman visualisasi peta kerjasama peneliti. Pada Kode Sumber 4. 5 kelas View terdiri dari file HTML, JavaScript, dan CSS.

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/html">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="target-
densitydpi=device-dpi, width=device-width, initial-
scale=1.0, maximum-scale=1">
    <meta name="description" content="Resources
ITS">
```

```

<meta name="author" content="BTST ITS">
<meta name="keywords" content="ITS, Resource,
windows 8, modern style, Metro UI, style, modern,
css, framework">

<link rel="stylesheet" href="<?php echo
base_url();?>assets/css/autocomplete/jquery-ui-
1.10.4.css" type="text/css" media="screen"/>
<script type="text/javascript" src="<?php echo
base_url();?>assets/js/autocomplete/jquery-
1.10.2.js"></script>

<title>Resources ITS</title>
<style>
    body {
        font: 10pt comic sans;
    }
    #mygraph {
        width: 100%;
        height: 620px;
        border: 1px solid lightgray;
    }
</style>

```

Kode Sumber 4. 5 Implementasi View Tema Header Halaman Visualisasi Peta Kerjasama Peneliti

4.3.3 Implementasi Controller

Kelas Controller merupakan penghubung antara kelas View dan kelas Model. Di dalam Controller terdapat fungsi – fungsi yang memproses permintaan dari View ke dalam struktur data di dalam Model. Pada kaskas visualisasi peta kerjasama peneliti terdapat beberapa kelas Controller. Kode Sumber 4. 6 merupakan implementasi dari kelas Controller Manage. Kelas Controller yang menjembatani komunikasi antara kelas Model dan kelas View memiliki fungsi yang menjembatani hal tersebut. Bila kelas View membutuhkan informasi dari basis data maka tugas dari Controller

untuk mengolah permintaan data tersebut dan meminta dari kelas Controller untuk mengirimkan kembali data yang dibutuhkan dari kelas View. Pada kelas Controller pada kerangka kerja CodeIgniter ini juga dapat menginisiasi *library* dan Model yang dibutuhkan seperti pada Kode Sumber 4. 6.

```
<?php
class Manage extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $this->load->model('index_model');
        $this->load->library('zend');
        $this->zend->load('Zend/Search/Lucene');

        $this->search_index = APPPATH .
'search/experts';

        Zend_Search_Lucene::setDefaultSearchField('all');
        $this->load->helper('url');
        ini_set('max_execution_time', 0);
        ini_set('memory_limit', '1024M');
    }
}
```

Kode Sumber 4. 6 Implementasi Kelas Controller Manage

4.4 Implementasi Basis Data

Pada subbab ini akan dibahas implementasi dari rancangan basis data yang telah dibahas pada Bab 3. Terdapat dua implementasi pada subbab ini yakni implementasi struktur basis data dan implementasi *query* yang digunakan kakas visualisasi peta kerjasama peneliti.

4.4.1. Implementasi Struktur Basis Data

Implementasi struktur basis data merupakan implementasi sintaks yang digunakan untuk membangun tabel-tabel yang dibutuhkan pada kakas visualisasi peta kerjasama peneliti.

Implementasi basis data menggunakan sintaks SQL dan terbagi menjadi beberapa tabel.

4.4.1.1. Implementasi Tabel Kata Dasar (tran_bobot_katadasar)

Tabel tran_bobot_katadasar digunakan untuk menyimpan data kata dasar bahasa indonesia. Kata dasar digunakan pada tahap *stemming* menggunakan algoritma Porter Bahasa Indonesia. Implementasi tabel tran_bobot_katadasar ditunjukkan pada Kode Sumber 4. 7.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_katadasar" (
    "id_katadasar" INT NOT NULL,
    "katadasar" VARCHAR(50) NOT NULL,
    "tipe_katadasar" VARCHAR(20) NOT NULL,
    PRIMARY KEY ("id_katadasar")
);
```

Kode Sumber 4. 7 Implementasi Struktur Basis Data Tabel tran_bobot_katadasar

4.4.1.2. Implementasi Tabel Kata Indeks (tran_bobot_kata_indeks)

Tabel tran_bobot_kata_indeks digunakan untuk menyimpan daftar *term* beserta pembobotannya. Daftar *term* tersebut merupakan daftar *term* yang sudah unik berasal dari *file* indeks yang sudah di *generate* menggunakan kerangka kerja Zend Search Lucene. Implementasi tabel tran_bobot_kata_indeks ditunjukkan pada Kode Sumber 4. 8.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_kata_indeks"
(
    "id_kata" INT NOT NULL,
    "kata" VARCHAR(255) NOT NULL,
    "df" INT NULL,
    "idf_norm" FLOAT NULL,
    PRIMARY KEY ("id_kata")
);
```



```
);
```

Kode Sumber 4. 8 Implementasi Struktur Basis Data Tabel tran_bobot_kata_indeks

4.4.1.3. Implementasi Tabel Kata Dokumen (tran_bobot_kata_dokumen)

Tabel tran_bobot_kata_dokumen digunakan untuk menyimpan daftar *term* beserta perhitungan bobot TF-IDF setiap *term* pada setiap dokumen. Implementasi tabel tran_bobot_kata_dokumen ditunjukkan pada Kode Sumber 4. 9.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_kata_dokumen"
(
    "id_dokumen" VARCHAR(50) NULL,
    "kata" VARCHAR(50) NULL,
    "tf" INT NULL,
    "tf_norm" FLOAT NULL,
    "tf_idf_norm" FLOAT NULL
);
```

Kode Sumber 4. 9 Implementasi Struktur Basis Data Tabel tran_bobot_kata_dokumen

4.4.1.4. Implementasi Tabel Vektor Topik (tran_bobot_vektor_topik)

Tabel tran_bobot_vektor_topik digunakan untuk menyimpan daftar *term* untuk setiap peneliti beserta pembobotan TF-IDF masing-masing *term*. Pada tabel tran_bobot_kata_dokumen daftar *term* disimpan dengan ID mahasiswa. Sedangkan pada tabel tran_bobot_vektor_topik ini setiap *term* disimpan menggunakan ID peneliti yang membimbing tugas akhir mahasiswa tersebut. Implementasi tabel tran_bobot_vektor_topik ditunjukkan pada Kode Sumber 4. 10.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_vektor_topik"
(
    "nidn" NCHAR(10) NOT NULL,
    "kata" VARCHAR(30) NULL,
```

```
"tf_idf_norm" FLOAT NULL
);
```

Kode Sumber 4. 10 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_topik

4.4.1.5. Implementasi Tabel Vektor 1 (tran_bobot_vektor_1)

Tabel tran_bobot_vektor_1 digunakan untuk menyimpan daftar *term* yang unik setiap peneliti beserta rata-rata TF-IDF *term* jika terdapat *term* yang sama pada dokumen berbeda. Dari daftar *term* setiap peneliti diambil beberapa *term* dengan bobot TF-IDF teratas, yang mana daftar *term* teratas inilah yang menjadi representasi topik penelitian setiap peneliti. Implementasi tabel tran_bobot_vektor_1 ditunjukkan pada Kode Sumber 4. 11.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_vektor_1" (
    "nidn" NCHAR(10) NULL,
    "kata" VARCHAR(50) NULL,
    "tf_idf_norm" FLOAT NULL
);
```

Kode Sumber 4. 11 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_1

4.4.1.6. Implementasi Tabel Vektor 2 (tran_bobot_vektor_2)

Tabel tran_bobot_vektor_2 digunakan untuk menyimpan bobot topik antar peneliti. Bobot topik merupakan hasil perhitungan dari daftar *term* kedua peneliti yang merepresentasikan kemiripan topik keduanya. Implementasi tabel tran_bobot_vektor_2 ditunjukkan pada Kode Sumber 4. 12.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_vektor_2" (
    "nidn1" NCHAR(10) NULL,
    "nidn2" NCHAR(10) NULL,
    "kali_tambah" FLOAT NULL,
    "kuadrat_akar_nidn1" FLOAT NULL,
```

```
"kuadrat_akar_nidn2" FLOAT NULL,
"final" FLOAT NULL
);
```

Kode Sumber 4. 12 Implementasi Struktur Basis Data Tabel tran_bobot_vektor_2

4.4.1.7. Implementasi Tabel Dosen Single (tran_bobot_dosen_single)

Tabel tran_bobot_dosen_single digunakan untuk menyimpan detail setiap peneliti. Implementasi tabel tran_bobot_dosen_single ditunjukkan pada Kode Sumber 4. 13.

```
CREATE TABLE IF NOT EXISTS "tran_bobot_dosen_single"
(
    "nidn" VARCHAR(50) NULL,
    "jumlah_dokumen" INT NULL,
    "id_node" INT NULL,
    "nip_baru" VARCHAR(50) NULL,
    "jumlah_kemiripan" INT NULL
);
```

Kode Sumber 4. 13 Implementasi Struktur Basis Data Tabel tran_bobot_dosen_single

4.4.1.8. Implementasi Tabel Dosen Peneliti (tran_bobot_dosen_peneliti)

Tabel tran_bobot_dosen_peneliti digunakan untuk menyimpan bobot hubungan kerjasama antar peneliti. Data yang disimpan meliputi bobot kerjasama penelitian yang dikerjakan secara bersama-sama dan bobot berdasarkan kemiripan topik. Hasil akhir bobot kerjasama peneliti merupakan hasil perkalian dari bobot kerjasama penelitian bersama dengan bobot kemiripan topik. Implementasi tabel tran_bobot_dosen_peneliti ditunjukkan pada Kode Sumber 4. 14.

```
CREATE TABLE IF NOT EXISTS
"tran_bobot_dosen_peneliti" (
    "nidn1" VARCHAR(50) NULL,
```

```
"nidn2" VARCHAR(50) NULL,
"bobot_peneliti" FLOAT NULL,
"bobot_topik" FLOAT NULL,
"final" FLOAT NULL
);
```

Kode Sumber 4. 14 Implementasi Struktur Basis Data Tabel tran_bobot_dosen_peneliti

4.4.1.9. Implementasi Tabel Admin (tref_user_admin)

Tabel tref_user_admin digunakan untuk menyimpan nama pengguna dan sandi administrator. Implementasi tabel tref_user_admin ditunjukkan pada Kode Sumber 4. 15.

```
CREATE TABLE IF NOT EXISTS "tref_user_admin" (
  "id" TINYINT NOT NULL,
  "username" VARCHAR(255) NOT NULL,
  "password" VARCHAR(255) NOT NULL,
  PRIMARY KEY ("id")
);
```

Kode Sumber 4. 15 Implementasi Struktur Basis Data Tabel tref_user_admin

4.4.2. Implementasi *Query*

Query yang digunakan pada aplikasi ini adalah SQL. Implementasi dari beberapa proses query akan dibahas dalam Sub bab berikut ini.

4.4.2.1. Implementasi Query Mengosongkan Semua Tabel Penghitungan Peta Kerjasama Peneliti

Berikut ini implementasi *query* untuk mengosongkan semua table yang digunakan untuk keperluan penghitungan peta kerjasama peneliti, seperti pada Kode Sumber 4. 16. Kode Sumber 4. 16 merupakan implementasi dari *query* untuk mengosongkan semua table yang digunakan untuk keperluan penghitungan peta kerjasama peneliti.

```

function process_empty_all()
{
    $sql = "delete from tran_bobot_dosen_peneliti
delete from tran_bobot_dosen_single
delete from tran_bobot_kata_dokumen
delete from tran_bobot_kata_indeks
delete from tran_bobot_vektor_1
delete from tran_bobot_vektor_2
delete from tran_bobot_vektor_topik
DBCC CHECKIDENT ('tran_bobot_kata_indeks',
RESEED, 0)
";
    $query = $this->db->query($sql);
    if($query)
    {
        return true;
    }
    else return false;
}

```

Kode Sumber 4. 16 Implementasi Query Mengosongkan Semua Tabel Penghitungan Peta Kerjasama Peneliti

4.4.2.2. Implementasi Query Mendapatkan Jumlah Dokumen Tugas Akhir

Berikut ini implementasi *query* untuk mendapatkan jumlah dokumen tugas akhir mahasiswa yang minimal memiliki satu dosen pembimbing, seperti pada Kode Sumber 4. 17. Kode Sumber 4. 17 merupakan implementasi dari *query* untuk mendapatkan jumlah dokumen tugas akhir mahasiswa yang minimal memiliki satu dosen pembimbing.

```

function get_rows_tmst_mahasiswa()
{
    $sql = "select nidn_promotor from tmst_mahasiswa
where nidn_promotor!='null'";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->num_rows();
    }
}

```

```

    else return null;
}

```

**Kode Sumber 4. 17 Implementasi Query Mendapatkan
Jumlah Dokumen Tugas Akhir**

4.4.2.3. Implementasi Query Mendapatkan Data Tugas Akhir

Berikut ini implementasi *query* untuk mendapatkan data dokumen tugas akhir dari setiap mahasiswa yang didalamnya juga terdapat ID dosen/peneliti pembimbing, seperti pada Kode Sumber 4. 18. Kode Sumber 4. 18 merupakan implementasi dari *query* untuk mendapatkan data dokumen tugas akhir dari setiap mahasiswa yang didalamnya juga terdapat ID dosen/peneliti pembimbing.

```

function
get_student_thesis_with_limit($perPage, $uri)
{
    $this->db->limit($perPage, $uri);
    $this->db->
>select('nim,nidn_promotor,nidn_kopromotor1,nidn_kop
romotor2,Abstrak_skripsi_id,Katakunci_skripsi,Judul_
skripsi_id_penuh,jalur_skripsi');
    $this->db->where('nidn_promotor!','=','null');
    $query = $this->db->get('tmst_mahasiswa');
    if($query->num_rows() > 0)
    {
        return $query->result_array();
    }
    else
    return null;
}

```

**Kode Sumber 4. 18 Implementasi Query Mendapatkan Data
Tugas Akhir**

4.4.2.4. Implementasi Query Mendapatkan Kata Dasar

Berikut ini implementasi *query* untuk mendapatkan kata dasar yang digunakan pada tahap *stemming*, seperti pada Kode Sumber 4. 19. Kode Sumber 4. 19 merupakan implementasi dari *query* untuk mendapatkan kata dasar yang digunakan pada tahap *stemming*.

```
function get_kata_dasar($kata)
{
    $sql = "SELECT * FROM tran_bobot_katadasar WHERE
katadasar='$kata'";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return true;
    }
    else return false;
}
```

Kode Sumber 4. 19 Implementasi Query Mendapatkan Kata Dasar

4.4.2.5. Implementasi Query Menyimpan Daftar Term Unik

Berikut ini implementasi *query* untuk menyimpan daftar *term* unik beserta frekuensi munculnya untuk seluruh dokumen, seperti pada Kode Sumber 4. 20. Kode Sumber 4. 20 merupakan implementasi dari *query* untuk menyimpan daftar *term* unik beserta frekuensi munculnya untuk seluruh dokumen.

```
function insert_tran_bobot_kata_indeks($kata,$df,
$dfnorm)
{
    $data = array(
        'kata' => $kata,
        'df' => $df,
        'df_norm' => $dfnorm
    );
}
```

```

    $status=$this->db-
>insert('tran_bobot_kata_indeks', $data);
    if($status)
    {
        return true;
    }
    else
    return false;
}

```

Kode Sumber 4. 20 Implementasi Query Menyimpan Daftar Term Unik

4.4.2.6. Implementasi Query Mendapatkan Daftar Term Unik Berdasarkan Frekuensi Kemunculan

Berikut ini implementasi *query* untuk mendapatkan daftar *term* unik berdasarkan frekuensi kemunculannya. Jika frekuensi munculnya *term* terlalu sedikit, maka kemungkinan *term* tersebut adalah mengalami kesalahan penulisan, sehingga tidak diambil. Sebaliknya jika frekuensi munculnya *term* terlalu banyak, maka *term* tersebut dianggap tidak penting atau tidak spesifik pada topik tertentu. Kode sumber seperti memasukkan daftar *term* unik berdasarkan frekuensi kemunculannya dapat dilihat pada Kode Sumber 4. 21. Kode Sumber 4. 21 merupakan implementasi dari *query* untuk mendapatkan daftar *term* unik berdasarkan frekuensi kemunculannya.

```

function
get_tran_bobot_kata_indeks_for_kata_dokumen()
{
    $sql = "
select cast(round((select count(*) from
tmst_mahasiswa where nidn_promotor!='null') *
0.0005,0) as float) as hitung";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {

```



```

        foreach ($query->result() as $row)
        {
            $bawah=$row->hitung;
        }
    }
    $sql = "
select cast(round((select count(*) from
tmst_mahasiswa where nidn_promotor!='null') * 0.1,0)
as float) as hitung";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        foreach ($query->result() as $row)
        {
            $atas=$row->hitung;
        }
    }
    $sql = "select * from tran_bobot_kata_indeks
where df >".$bawah." and df <".$atas." ";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->result();
    }
    else
    return null;
}

```

Kode Sumber 4. 21 Implementasi Query Mendapatkan Daftar Term Unik Berdasarkan Frekuensi Kemunculan

4.4.2.7. Implementasi Query Menyimpan Daftar Term Beserta Frekuensi Munculnya Untuk Setiap Dokumen

Berikut ini implementasi *query* untuk menyimpan daftar *term* beserta frekuensi munculnya untuk setiap dokumen, seperti pada Kode Sumber 4. 22. Kode Sumber 4. 22 merupakan implementasi dari *query* untuk menyimpan daftar *term* beserta frekuensi munculnya untuk setiap dokumen.

```

function
insert_tran_bobot_kata_dokumen($id_dok,$kata,$tf)
{
    $data = array(
        'id_dokumen' => $id_dok,
        'kata' => $kata,
        'tf' => $tf
    );

    $status=$this->db->
>insert('tran_bobot_kata_dokumen', $data);
    if($status)
    {
        return true;
    }
    else
    return false;
}

```

Kode Sumber 4. 22 Implementasi Query Menyimpan Daftar Term Beserta Frekuensi Munculnya Untuk Setiap Dokumen

4.4.2.8. Implementasi Query Memperbarui TF-IDF Setiap Term

Berikut ini implementasi *query* untuk memperbarui TF-IDF setiap *term*, seperti pada Kode Sumber 4. 23. Kode Sumber 4. 23 merupakan implementasi dari *query* untuk memperbarui TF-IDF setiap *term*.

```

function process_tfnorm_tfidfnorm()
{
    $sql = "UPDATE tran_bobot_kata_dokumen
        SET tf_norm =0.5 + 0.5 * (tf / (1+
        (CONVERT(DECIMAL(9,2), (SELECT MAX(tf) as tf_max
        FROM (select * from
        tran_bobot_kata_dokumen as x
        where
        id_dokumen=tran_bobot_kata_dokumen.id_dokumen)
        b))))))
        FROM tran_bobot_kata_indeks

```

```

WHERE
    tran_bobot_kata_indeks.kata =
tran_bobot_kata_dokumen.kata

UPDATE tran_bobot_kata_dokumen
SET tf_idf_norm =
tran_bobot_kata_indeks.idf_norm *tf_norm
FROM tran_bobot_kata_indeks
WHERE
    tran_bobot_kata_indeks.kata =
tran_bobot_kata_dokumen.kata
";
$query = $this->db->query($sql);
if($query)
{
    return true;
}
else return false;
}

```

Kode Sumber 4. 23 Implementasi Query Memperbarui TF-IDF Setiap Term

4.4.2.9. Implementasi Query Menyimpan Term Berdasarkan ID Peneliti

Berikut ini implementasi *query* untuk menyimpan *term* berdasarkan ID peneliti, seperti pada Kode Sumber 4. 24. Kode Sumber 4. 24 merupakan implementasi dari *query* untuk menyimpan *term* berdasarkan ID peneliti.

```

function process_tfidfnorm_with_expert_id()
{
    $sql = "insert into tran_bobot_vektor_topik
select mhs.nidn_promotor as nidn, dok.kata as
kata, dok.tf_idf_norm as tf_idf_norm
from tran_bobot_kata_dokumen dok, tmst_mahasiswa
mhs
where dok.id_dokumen=mhs.nim and
mhs.nidn_promotor!='null'

insert into tran_bobot_vektor_topik

```

```

select mhs.nidn_kopromotor1 as nidn, dok.kata as
kata, dok.tf_idf_norm as tf_idf_norm
from tran_bobot_kata_dokumen dok, tmst_mahasiswa
mhs
where dok.id_dokumen=mhs.nim and
mhs.nidn_kopromotor1!='null'

insert into tran_bobot_vektor_topik
select mhs.nidn_kopromotor2 as nidn, dok.kata as
kata, dok.tf_idf_norm as tf_idf_norm
from tran_bobot_kata_dokumen dok, tmst_mahasiswa
mhs
where dok.id_dokumen=mhs.nim and
mhs.nidn_kopromotor2!='null'
";
$query = $this->db->query($sql);
if($query)
{
    return true;
}
else return false;
}

```

Kode Sumber 4. 24 Implementasi Query Menyimpan Term Berdasarkan ID Peneliti

4.4.2.10. Implementasi Query Menyimpan Daftar Term Teratas Setiap Peneliti

Berikut ini implementasi *query* untuk menyimpan daftar *term* teratas setiap peneliti, seperti pada Kode Sumber 4. 25. Kode Sumber 4. 25 merupakan implementasi dari *query* untuk menyimpan daftar *term* teratas setiap peneliti.

```

function insert_tran_bobot_vektor_1($nidn)
{
    $sql = "insert into tran_bobot_vektor_1
select top 150 '". $nidn. "', kata,
avg(tf_idf_norm) as tf_idf_norm
from tran_bobot_vektor_topik
where nidn='". $nidn. "'
group by kata

```

```

order by tf_idf_norm desc";
$status = $this->db->query($sql);
if($status)
{
    return true;
}
else
return false;
}

```

Kode Sumber 4. 25 Implementasi Query Menyimpan Daftar Term Teratas Setiap Peneliti

4.4.2.11. Implementasi Query Mendapatkan Daftar Term Semua Peneliti

Berikut ini implementasi *query* untuk mendapatkan daftar *term* semua peneliti, seperti pada Kode Sumber 4. 26. Kode Sumber 4. 26 merupakan implementasi dari *query* untuk mendapatkan daftar *term* semua peneliti.

```

function get_tran_bobot_vektor_1()
{
    $sql = "select * from tran_bobot_vektor_1";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->result_array();
    }
    else
    return null;
}

```

Kode Sumber 4. 26 Implementasi Query Mendapatkan Daftar Term Semua Peneliti

4.4.2.12. Implementasi Query Menyimpan Hubungan Antar Peneliti Berdasarkan ID Peneliti

Berikut ini implementasi *query* untuk menyimpan hubungan antar peneliti berdasarkan ID peneliti, seperti pada Kode Sumber

4. 27. Kode Sumber 4. 27 merupakan implementasi dari *query* untuk menyimpan hubungan antar peneliti berdasarkan ID peneliti.

```
function process_expert_topics_relation_with_id()
{
    $sql = "insert into tran_bobot_dosen_peneliti
    SELECT a.nidn as nidn1,b.nidn as nidn2,0,0,0
    FROM (select distinct(nidn) from
    tran_bobot_vektor_topik) a
    JOIN (select distinct(nidn) from
    tran_bobot_vektor_topik) b
    ON a.nidn < b.nidn

    insert into tran_bobot_vektor_2
    SELECT a.nidn as nidn1,b.nidn as nidn2,0,0,0,0
    FROM (select distinct(nidn) from
    tran_bobot_vektor_topik) a
    JOIN (select distinct(nidn) from
    tran_bobot_vektor_topik) b
    ON a.nidn < b.nidn'
    ";
    $query = $this->db->query($sql);
    if($query)
    {
        return true;
    }
    else return false;
}
```

Kode Sumber 4. 27 Implementasi Query Menyimpan Hubungan Antar Peneliti Berdasarkan ID Peneliti

4.4.2.13. Implementasi Query Mendapatkan Hubungan Antar Peneliti Berdasarkan ID Peneliti

Berikut ini implementasi *query* untuk mendapatkan hubungan antar peneliti berdasarkan ID peneliti, seperti pada Kode Sumber 4. 28. Kode Sumber 4. 28 merupakan implementasi dari *query* untuk mendapatkan hubungan antar peneliti berdasarkan ID peneliti.

```

function get_nidn_vektor_double()
{
    $sql = "select nidn1,nidn2 from
tran_bobot_dosen_peneliti";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->result_array();
    }
    else
    return null;
}

```

Kode Sumber 4. 28 Implementasi Query Mendapatkan Hubungan Antar Peneliti Berdasarkan ID Peneliti

4.4.2.14. Implementasi Query Memperbarui Hubungan Bobot Topik Antar Peneliti

Berikut ini implementasi *query* untuk memperbarui hubungan bobot topik antar peneliti, seperti pada Kode Sumber 4. 29. Kode Sumber 4. 29 merupakan implementasi dari *query* untuk memperbarui hubungan bobot topik antar peneliti.

```

function
update_tran_bobot_vektor_2($nidn1,$nidn2,$sumatas,$sumkiri,$sumkanan,$sumfinal)
{
    $data = array(
        'kali_tambah' => $sumatas,
        'kuadrat_akar_nidn1' => $sumkiri,
        'kuadrat_akar_nidn2' => $sumkanan,
        'final' => $sumfinal
    );
    $this->db->where('nidn1', $nidn1);
    $this->db->where('nidn2', $nidn2);
    $status=$this->db->update('tran_bobot_vektor_2',
    $data);
    if($status)
    {
        return true;
    }
    else

```

```

    return false;
}

```

Kode Sumber 4. 29 Implementasi Query Memperbarui Hubungan Bobot Topik Antar Peneliti

4.4.2.15. Implementasi Query Memperbarui Hubungan Bobot Kerjasama Antar Peneliti

Berikut ini implementasi *query* untuk memperbarui hubungan bobot kerjasama antar peneliti, seperti pada Kode Sumber 4. 30. Kode Sumber 4. 30 merupakan implementasi dari *query* untuk memperbarui hubungan bobot kerjasama antar peneliti.

```

function
update_tran_bobot_dosen_peneliti($n1,$n2,$b)
{
    $data = array(
        'bobot_peneliti' => $b
    );
    $array1 = array('nidn1' => $n1, 'nidn2'
=> $n2);
    $this->db->where($array1);

    $status=$this->db-
>update('tran_bobot_dosen_peneliti', $data);
    if($status)
    {
        return true;
    }
    else
    return false;
}

```

Kode Sumber 4. 30 Implementasi Query Memperbarui Hubungan Bobot Kerjasama Antar Peneliti

4.4.2.16. Implementasi Query Memperbarui Bobot Final Hubungan Antar Peneliti

Berikut ini implementasi *query* untuk memperbarui bobot final hubungan antar peneliti, seperti pada Kode Sumber 4. 31.

Bobot final merupakan representasi kedekatan kerjasama berdasarkan kemiripan topik antar peneliti. Kode Sumber 4. 31 merupakan implementasi dari *query* untuk memperbarui bobot final hubungan antar peneliti.

```
function process_expert_final_relation()
{
    $sql = "UPDATE tran_bobot_dosen_peneliti
    SET bobot_topik = tran_bobot_vektor_2.final,
    final=tran_bobot_vektor_2.final*bobot_peneliti
    FROM tran_bobot_vektor_2
    WHERE
        tran_bobot_dosen_peneliti.nidn1 =
        tran_bobot_vektor_2.nidn1 and
        tran_bobot_dosen_peneliti.nidn2 =
        tran_bobot_vektor_2.nidn2
    ";
    $query = $this->db->query($sql);
    if($query)
    {
        return true;
    }
    else return false;
}
```

Kode Sumber 4. 31 Implementasi Query Memperbarui Bobot Final Hubungan Antar Peneliti

4.4.2.17. Implementasi Query Mendapatkan Data Setiap Peneliti

Berikut ini implementasi *query* untuk mendapatkan data setiap peneliti, seperti pada Kode Sumber 4. 32. Kode Sumber 4. 32 merupakan implementasi dari *query* untuk mendapatkan data setiap peneliti.

```
function get_nidn_vektor_with_nip()
{
    $sql = "select
    distinct(tran_bobot_vektor_topik.nidn) as
    nidn unique, tmst_dosen.*, tran_bobot_dosen single.*
```

```

from tran_bobot_vektor_topik, tmst_dosen,
tran_bobot_dosen_single,
(select *
from tran_bobot_dosen_peneliti
where final>0.01) as xc
where tran_bobot_vektor_topik.nidn=tmst_dosen.nidn
and tran_bobot_dosen_single.nidn=tmst_dosen.nidn and
(tmst_dosen.nidn=xc.nidn1 or
tmst_dosen.nidn=xc.nidn2)";
$query = $this->db->query($sql);
if($query->num_rows() > 0)
{
    return $query->result_array();
}
else
return null;
}

```

Kode Sumber 4. 32 Implementasi Query Mendapatkan Data Setiap Peneliti

4.4.2.18. Implementasi Query Mendapatkan Fakultas Peneliti

Berikut ini implementasi *query* untuk mendapatkan fakultas peneliti, seperti pada Kode Sumber 4. 33. Kode Sumber 4. 33 merupakan implementasi dari *query* untuk mendapatkan fakultas peneliti.

```

function get_fakultas_dosen_peneliti($id)
{
    $sql = "select nama_fakultas_en from
tmst_fakultas where
kode_fakultas='$id'";
$query = $this->db->query($sql);
if($query->num_rows() > 0)
{
    foreach ($query->result() as $row)
    {
        return $row->nama_fakultas_en;
    }
}
}

```

```

else
return null;
}

```

Kode Sumber 4. 33 Implementasi Query Mendapatkan Fakultas Peneliti

4.4.2.19. Implementasi Query Mendapatkan Jurusan Peneliti

Berikut ini implementasi *query* untuk mendapatkan jurusan peneliti, seperti pada Kode Sumber 4. 34. Kode Sumber 4. 34 merupakan implementasi dari *query* untuk mendapatkan jurusan peneliti.

```

function get_jurusan_dosen_peneliti($kode)
{
    $sql = "select Nama_jurusan_en from tmst_jurusan
           where Kode_jurusan='$kode'";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        foreach ($query->result() as $row)
        {
            return $row->Nama_jurusan_en;
        }
    }
    else return null;
}

```

Kode Sumber 4. 34 Implementasi Query Mendapatkan Jurusan Peneliti

4.4.2.20. Implementasi Query Memperbarui ID Node Peneliti

Berikut ini implementasi *query* untuk memperbarui ID *node* peneliti, seperti pada Kode Sumber 4. 35. Kode Sumber 4. 35 merupakan implementasi dari *query* untuk memperbarui ID *node* peneliti.

```

function
update_tran_bobot_dosen_single($nidn,$id,$nip)

```

```

{
    $data = array(
        'id_node' => $id,
        'nip_baru' => $nip
    );
    $this->db->where('nidn', $nidn);
    $status=$this->db-
>update('tran_bobot_dosen_single', $data);
    if($status)
    {
        return true;
    }
    else
    return false;
}

```

Kode Sumber 4. 35 Implementasi Query Memperbarui ID Node Peneliti

4.4.2.21. Implementasi Query Memperbarui Jumlah Peneliti yang Mirip untuk Setiap Peneliti

Berikut ini implementasi *query* untuk memperbarui jumlah peneliti yang mirip untuk setiap peneliti, seperti pada Kode Sumber 4. 36. Kode Sumber 4. 36 merupakan implementasi dari *query* untuk memperbarui jumlah peneliti yang mirip untuk setiap peneliti.

```

function process_expert_similarity_update()
{
    $sql = "update tran_bobot_dosen_single
    set jumlah_kemiripan=de.jumlah
    from (
        select count(tran_bobot_dosen_single.nidn)
as jumlah, tran_bobot_dosen_single.nidn
    from tran_bobot_dosen_single,
        (select *
        from tran_bobot_dosen_peneliti
        where final>0.01) as xc
        where (tran_bobot_dosen_single.nidn=xc.nidn1
or tran_bobot_dosen_single.nidn=xc.nidn2)
    group by tran_bobot_dosen_single.nidn )as de

```

```

        where tran_bobot_dosen_single.nidn = de.nidn
        ";
$query = $this->db->query($sql);
if($query)
{
    return true;
}
else return false;
}

```

Kode Sumber 4. 36 Implementasi Query Memperbarui Jumlah Peneliti yang Mirip untuk Setiap Peneliti

4.4.2.22. Implementasi Query Mendapatkan Bobot Hubungan Kerjasama Berdasarkan Kemiripan Topik Penelitian Antar Peneliti

Berikut ini implementasi *query* untuk mendapatkan bobot hubungan kerjasama berdasarkan kemiripan topik penelitian antar peneliti, seperti pada Kode Sumber 4. 37. Kode Sumber 4. 37 merupakan implementasi dari *query* untuk mendapatkan bobot hubungan kerjasama berdasarkan kemiripan topik penelitian antar peneliti.

```

function get_tran_bobot_dosen_peneliti()
{
    $sql = "select distinct *
from tran_bobot_dosen_peneliti
where final>0.009";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->result_array();
    }
    else
    return null;
}

```

Kode Sumber 4. 37 Implementasi Query Mendapatkan Bobot Hubungan Kerjasama Berdasarkan Kemiripan Topik Penelitian Antar Peneliti

4.4.2.23. Implementasi Query Mendapatkan Bobot Hubungan Antar Peneliti Paling Tinggi

Berikut ini implementasi *query* untuk mendapatkan bobot hubungan antar peneliti paling tinggi, seperti pada Kode Sumber 4. 38. Kode Sumber 4. 38 merupakan implementasi dari *query* untuk mendapatkan bobot hubungan antar peneliti paling tinggi.

```
function get_max_final_tran_bobot_dosen_peneliti()
{
    $sql = "select max(final) as max
from tran_bobot_dosen_peneliti
where final>0.009";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        foreach ($query->result() as $row)
        {
            return $row->max;
        }
    }
    else
    return null;
}
```

Kode Sumber 4. 38 Implementasi Query Mendapatkan Bobot Hubungan Antar Peneliti Paling Tinggi

Pada Kode Sumber 4. 37 dan Kode Sumber 4. 38 terlihat bahwa *threshold* bobot hubungan antar peneliti adalah 0,009. Batas *threshold* digunakan agar visualisasi lebih informatif dengan jumlah *edge* (hubungan) yang ditampilkan pada graf visualisasi. Batas ini ditentukan berdasarkan hasil survei ke pengguna seperti pada Lampiran D Tabel D. 1 tentang penentuan *threshold* bobot hubungan kerjasama antar peneliti.

4.4.2.24. Implementasi Query Mendapatkan Node Peneliti

Berikut ini implementasi *query* untuk mendapatkan *node* peneliti, seperti pada Kode Sumber 4. 39. Kode Sumber 4. 39 merupakan implementasi dari *query* untuk mendapatkan *node* peneliti.

```
function get_node_tran_bobot_dosen_single($nidn)
{
    $sql = "select * from tran_bobot_dosen_single,
tmst_dosen where tmst_dosen.nidn='".$nidn.'" and
tran_bobot_dosen_single.nidn='".$nidn.'";
    $query = $this->db->query($sql);
    if($query->num_rows() > 0)
    {
        return $query->result_array();
    }
    else {
        return null;
    }
}
```

Kode Sumber 4. 39 Implementasi Query Mendapatkan Node Peneliti

4.5 Implementasi Proses Aplikasi

Pada subbab ini akan dibahas berbagai fungsi atau proses yang berjalan pada kakas visualisasi peta kerjasama peneliti. Proses baik yang dilakukan karena permintaan pengguna maupun proses yang secara otomatis dilakukan aplikasi tanpa perlu adanya permintaan dari pengguna.

4.5.1 Proses Pengindeksan Data Penelitian Tugas Akhir

Berikut ini implementasi dari proses pengindeksan data penelitian tugas akhir. Proses pengindeksan data penelitian tugas akhir dilakukan ketika ingin membuat atau memperbarui data penelitian tugas akhir yang diproses pada visualisasi kerjasama

peneliti. Kode Sumber 4. 40 merupakan kode untuk melakukan pengindeksan data penelitian tugas akhir.

```
function indexing_publics()
{
    $count=0;
    $limit = 1000;
    $start = 0;
    $length=$this->index_model->get_rows_tmst_mahasiswa();
    $limit=$length;

    Zend_Search_Lucene_Analysis_Analyzer::setDefault(new
    Zend_Search_Lucene_Analysis_Analyzer_Common_TextNum_
    CaseInsensitive());
    $index = Zend_Search_Lucene::create(APPPATH
    . 'search/visualizations');
    $no=0;
    ...
    $index->commit();
    $this->optimize(APPPATH .
    'search/visualizations');
}
```

Kode Sumber 4. 40 Kode Sumber Proses Pengindeksan Data Penelitian Tugas Akhir

4.5.2 Proses Tokenisasi dan Menghilangkan Stopwords

Berikut ini implementasi proses proses menghilangkan *stopwords*, seperti pada Kode Sumber 4. 41. *Stopwords* dihilangkan ketika sudah didapatkan daftar *term* unik. *Term* yang termasuk *stopwords* dihilangkan. Kode Sumber 4. 41 merupakan implementasi proses proses menghilangkan *stopwords*.

```
$index = Zend_Search_Lucene::open(APPPATH .
'search/visualizations');
    $result=$index->terms();
    $terms=array();
    $x=0;
    $y=count($result);
```



```

        while ($y--)
        {
            if(preg_match('#[0-9]#',$result[$x]-
>text)==false&&strlen($result[$x]-
>text)>2&&strlen($result[$x]->text)<=25)
                $terms[$x]=$result[$x]->text;
                $x++;
        }
        $stop="akibat, alasan, alasannya,
akibatnya, ... , yaitu, yakin, yakni, yang";
        $string = preg_replace('/\s+/', ' ',
$stop);
        $stops= preg_split('/[\\s,]+/', $string);

        $terms =
array_merge(array_diff($terms,$stops));
        ...
    }

```

Kode Sumber 4. 41 Kode Sumber Proses Menghilangkan Stopwords

4.5.3 Proses Stemming

Berikut ini implementasi dari proses *stemming*. Kode Sumber 4. 42 merupakan kode untuk melakukan proses *stemming*.

```

function insert_tran_bobot_kata_indeks()
{
    ...
    $terms = array_merge(array_diff($terms,$stops));
    foreach($terms as $t)
    {
        $t=$this->wordStemming($t);
    }
    ...
}
//Stemming
function wordStemming($kataAwal) { ... }
function cari($kataAwal) { ... }
//langkah 1 - hapus partikel
function hapuspartikel($kata){ ... }
//langkah 2 - hapus possessive pronoun
function hapuspp($kata) { ... }

```

```
//langkah 3 hapus first order prefiks (awalan
pertama)
function hapusawalan1($kata){ ... } //langkah 4
hapus second order prefiks (awalan kedua)
function hapusawalan2($kata){ ... } ///langkah 5
hapus suffiks
function hapusakhiran($kata){ ... }
```

Kode Sumber 4. 42 Kode Sumber Proses Melakukan Stemming

4.5.4 Proses Pembobotan Kata (Term)

Berikut ini implementasi dari proses pembobotan kata (*term*). Proses pembobotan kata dilakukan untuk mendapatkan daftar kata penting dari setiap peneliti. Kode Sumber 4. 43 merupakan kode untuk pembobotan kata.

```
function insert_tran_bobot_kata_indeks()
{
    ...
    $terms=array_unique($terms);
    $termsfdocreq=array();
    foreach($terms as $t)
    {
        $termsfdocreq[$t]=$this->
term_doc_freq($t);
    }
    ...
}
function term_doc_freq($t){ ... }
function insert_tran_bobot_kata_dokumen(){ ... }
function process_tfnorm_tfidfnorm(){ ... }
function process tfidfnorm with expert id{ ... }
function
process_expert_topics_relation_with_id{ ... }
function insert_tran_bobot_vektor_1{ ... }
```

Kode Sumber 4. 43 Kode Sumber Proses Pembobotan Kata

4.5.5 Proses Penghitungan Bobot Topik Antar Peneliti

Berikut ini implementasi dari proses penghitungan bobot relasi topik antar peneliti. Proses penghitungan bobot relasi topik antar peneliti dilakukan untuk mendapatkan kemiripan topik penelitian antar peneliti. Kode Sumber 4. 44 merupakan kode untuk penghitungan bobot relasi topik antar peneliti.

```
function update_tran_bobot_vektor_2()
{
    $vls=$this->index_model-
>get_tran_bobot_vektor_1();
    $vektors= array(array());
    foreach($vls as $v1)
    {
        $nidn=$v1['nidn'];
        $kata=$v1['kata'];
        $tf_idf_norm=$v1['tf_idf_norm'];

        $vektors[$nidn][$kata]=$tf_idf_norm;
    }
    $nidns=$this->index_model-
>get_nidn_vektor_double();
    foreach($nidns as $ni)
    {
        ...
    }
}
```

Kode Sumber 4. 44 Kode Sumber Proses Penghitungan Bobot Relasi Topik Antar Peneliti

4.5.6 Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti

Berikut ini implementasi dari proses penghitungan bobot kerjasama penelitian. Proses penghitungan bobot kerjasama penelitian dilakukan untuk mendapatkan kedekatan peneliti berdasarkan penelitian tugas akhir yang dilakukan secara bersama. Kode Sumber 4. 45 merupakan kode untuk penghitungan bobot kerjasama penelitian.

```

function update_tran_bobot_dosen_peneliti()
{
    $index =
Zend_Search_Lucene::open(APPPATH .
'search/visualizations');
    $query_result = $index-
>find('all:"yes"');
    $expert=array();
    $expwork=array();
    $i=0;
    $it=0;
    $ik=0;
    foreach($query_result as $q)
    {

        $temp=$query_result[$it];
        if($temp->nidn1){ $expert[$i]=
$temp->nidn1; $i++;}
        if($temp->nidn2){ $expert[$i]=
$temp->nidn2; $i++;}
        if($temp->nidn3){ $expert[$i]=
$temp->nidn3; $i++;}

        ...
    }
}

```

Kode Sumber 4. 45 Kode Sumber Proses Penghitungan Bobot Kerjasama Penelitian Antar Peneliti

4.5.7 Proses Penghitungan Bobot Kerjasama Berdasarkan Topik Penelitian Antar Peneliti

Berikut ini implementasi dari proses penghitungan bobot kerjasama berdasarkan topik penelitian antar peneliti. Proses penghitungan bobot kerjasama berdasarkan topik penelitian antar peneliti dilakukan untuk mendapatkan hubungan kerjasama peneliti berdasarkan topik penelitian antar peneliti. Kode Sumber 4. 46 merupakan kode untuk penghitungan bobot kerjasama berdasarkan topik penelitian antar peneliti.

```

function process_expert_final_relation()

```

```

        {
            $sql = "UPDATE tran_bobot_dosen_peneliti
                SET bobot_topik =
tran_bobot_vektor_2.final,
final=tran_bobot_vektor_2.final*bobot_peneliti
                FROM tran_bobot_vektor_2
                WHERE
                    tran_bobot_dosen_peneliti.nidn1 =
tran_bobot_vektor_2.nidn1 and
                    tran_bobot_dosen_peneliti.nidn2 =
tran_bobot_vektor_2.nidn2
                ";
            $query = $this->db->query($sql);
            if($query)
            {
                return true;
            }
            else return false;
        }
    }

```

Kode Sumber 4. 46 Kode Sumber Proses Penghitungan Bobot Kerjasama Berdasarkan Topik Penelitian Antar Peneliti

4.5.8 Proses Pemodelan Graf Peta Kerjasama Peneliti Dalam Bentuk JSON

Berikut ini implementasi dari proses pemodelan graf peta kerjasama peneliti dalam bentuk JSON. Proses pemodelan graf dalam bentuk JSON dilakukan agar kakas visualisasi tidak mengakses secara langsung peta kerjasama peneliti ke data yang tersimpan didalam kamus data. Sehingga akan mengurangi sumber daya yang diperlukan ketika kakas visualisasi diakses. Kode Sumber 4. 47 merupakan pemodelan graf peta kerjasama peneliti dalam bentuk JSON.

```

function write_json_file()
{
    $nodes=$this->index_model-
>get_nidn_vektor_with_nip();
    $length=count($nodes);
    $string nodes="[";

```

```

        $ind=1;
        foreach ($nodes as $no)
        {
...
        }
        $string_nodes=$string_nodes."]";
if (file_exists (APPPATH.'search/json/nodes.json'
'))
{
unlink (APPPATH.'search/json/nodes.json');

if (!file_put_contents (APPPATH.'search/json/nod
es.json', $string_nodes, FILE_APPEND)) {
}
}
else

if (!file_put_contents (APPPATH.'search/json/nod
es.json', $string_nodes, FILE_APPEND)) {
}

        $this->index_model->
>process_expert_similarity_update();
        $edges=$this->index_model->
>get_tran_bobot_dosen_peneliti();
        $length=count($edges);
        $string_edges="[";
        $ind=1;
        $max=$this->index_model->
>get_max_final_tran_bobot_dosen_peneliti();
        foreach ($edges as $ed)
        {
...
        }
        $string_edges=$string_edges."]";

```

```

if(file_exists(APPPATH.'search/json/edges.json'
'))

{unlink(APPPATH.'search/json/edges.json');

if(!file_put_contents(APPPATH.'search/json/edges.json', $string_edges, FILE_APPEND)){

}

else
if(!file_put_contents(APPPATH.'search/json/edges.json', $string_edges, FILE_APPEND)){

}
}

```

**Kode Sumber 4. 47 Kode Sumber Proses Pemodelan Graf
Peta Kerjasama Peneliti Dalam Bentuk JSON**

4.5.9 Proses Visualisasi Graf Berbasis Web

4.5.9.1 Proses Inisiasi Visualisasi Graf Peta Kerjasama Peneliti

Berikut ini implementasi dari proses inisiasi visualisasi graf peta kerjasama peneliti. Kode Sumber 4. 48 merupakan kode untuk inisiasi visualisasi graf peta kerjasama peneliti.

```

if (window.XMLHttpRequest)
{
    // code for IE7+, Firefox, Chrome, Opera, Safari
    xmlhttp=new XMLHttpRequest();
}
else
{
    // code for IE6, IE5
    xmlhttp=new
ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.open("GET","<?php echo
base_url();
?>application/search/json/nodes.json",false);
xmlhttp.send();
xmlDoc=xmlhttp.responseText;
nodes = JSON.parse(xmlDoc);

```

```

        var availableTags=[];
        for(var i in nodes)
        {
            nodes[i].image=DIR+nodes[i].image;
            availableTags.push(nodes[i].label);
        }
        var color = 'green';
        xmlhttp.open("GET","<?php echo base_url();
?>application/search/json/edges.json",false);
        xmlhttp.send();
        xmlDoc=xmlhttp.responseText;
        edges = JSON.parse(xmlDoc);
        var container =
document.getElementById('mygraph');
        var options = {
            };

        ...
        var value = 178;
        var
        nodestemp2=nodes.slice(value,nodes.length);
        nodestemp=nodes.slice(0,value);
        edgestemp=null;
        edgestemp=edges;
        //alert(edges.length);
        //alert(edgestemp.length);
        $.each(nodestemp2, function(i, item) {
            edgestemp=$.map(edgestemp,
function(val) {
                return
val.from.toString() != item.id ? val : null;
            });
            edgestemp=$.map(edgestemp,
function(val) {
                return
val.to.toString() != item.id ? val : null;
            });
        });

        var data = {
            nodes: nodestemp,
            edges: edgestemp

```



```
};
graph = new vis.Graph(container, data,
options);
```

Kode Sumber 4. 48 Kode Sumber Proses Inisiasi Visualisasi Graf Peta Kerjasama Peneliti

Pada Kode Sumber 4. 48 dapat dilihat bahwa jumlah *node* yang ditampilkan pada halaman utama peta kerjasama peneliti adalah 178 *node* peneliti teratas jumlah kemiripan topiknya dengan peneliti lain dari seluruh peneliti yang ada di ITS. Hal ini dikarenakan jumlah *node* yang ditampilkan akan mempengaruhi keinformatifan visualisasi peta kerjasama peneliti. Batas ini ditentukan berdasarkan hasil survei ke pengguna seperti pada Lampiran D Tabel D. 2 tentang jumlah *node* pada graf visualiasi.

4.5.9.2 Proses Menampilkan Detail Peneliti dan Mengurutkan Peneliti-Peneliti yang Mirip Topik Penelitiannya

Berikut ini implementasi dari proses menampilkan detail peneliti dan mengurutkan peneliti-peneliti yang mirip topik penelitiannya. Kode Sumber 4. 49 menampilkan detail peneliti dan mengurutkan peneliti-peneliti yang mirip topik penelitiannya.

```
var txt=(properties.edges).toString();
var resed=txt.split(",");
var reseds=[];
for(var j in resed)
{
    var found = $.map(edges, function(val) {
        return val.id == resed[j] ?
val.length : null;
    });
    reseds.push(found[0]+" "+resed[j]);
}
reseds.sort();
var tempreseds=reseds.toString().split(",");
var txtdup=[];
for(var j in tempreseds)
```

```

        {
            var tempres=reseds[j].split(" ");
            txtdup.push(tempres[1]+" "+tempres[2]);
        }
        var res = txtdup.toString().replace(/,/g, '
');
        var nres=res.split(' ');
        nres=nres.filter(onlyUnique);
        var index
        =nres.indexOf(properties.nodes.toString());
        nres.splice(index,1);
        if(properties.edges.length==0)
        nres.length=0;
        var imgElement =
        document.getElementById('gam');
        imgElement.src =
        nodes[(properties.nodes)-1].image;

        imgElement.title=nodes[(properties.nodes)-1].title;
        imgElement.style.height= '166px';
        imgElement.style.width= '128px';
        imgElement.style.border='1px solid
        #2b5797';
        document.getElementById('name').innerHTML=nodes[(pro
        perties.nodes)-1].title;
        document.getElementById('documents').innerHTML="#doc
        uments: "+nodes[(properties.nodes)-1].value;
        document.getElementById('connections').innerHTML="#c
        onnections: "+nres.length;
        document.getElementById('department').innerHTML=node
        s[(properties.nodes)-1].department;
        document.getElementById('faculty').innerHTML=nodes[(
        properties.nodes)-1].faculty;
        $.get(temp_encrypt_url+nodes[(properties.nodes)-
        1].old_nip, function(data){

        document.getElementById('link').href=temp_base_url+d
        ata;

        });
        document.getElementById('detailinfo').innerHTML="Det
        ail Informations";
        $('#nonode').hide();

```

```

        $('#deselect').show();
        $('#detailxp').show();
        var
mydiv=document.getElementById('info');
        while(mydiv.firstChild)
        {
            mydiv.removeChild(mydiv.firstChild);
        }
        for(var i in nres)
        {
            var elem =document.createElement("img");
            elem.style.height= '83px';
            elem.style.width= '64px';
            elem.style.border='1px solid #2b5797';
            elem.id=(nres[i]).toString();
            elem.setAttribute("class", "dosen");
            elem.title=nodes[nres[i]-1].title;

            elem.src=nodes[nres[i]-1].image;
            document.getElementById('info').appendChild(elem);
        }

```

Kode Sumber 4. 49 Kode Sumber Proses Menampilkan Detail Peneliti dan Mengurutkan Peneliti-Peneliti yang Mirip Topik Penelitiannya

4.5.9.3 Proses Menghitung dan Menampilkan Persentase Persebaran Peneliti

Berikut ini implementasi dari proses menghitung dan menampilkan persentase persebaran peneliti. Kode Sumber 4. 50 merupakan kode untuk menghitung dan menampilkan persentase persebaran peneliti.

```

var founds = $.map(nodestemp, function(val) {
    return val.group == 'dept1' ? val.id
: null;
});
var dept1=founds.length;
var founds = $.map(nodestemp, function(val)
{

```

```

        return val.group == 'dept2' ? val.id
: null;
    });
    var dept2=founds.length;
    var founds = $.map(nodestemp, function(val)
{
        return val.group == 'dept3' ? val.id
: null;
    });
    var dept3=founds.length;
    var founds = $.map(nodestemp, function(val)
{
        return val.group == 'dept4' ? val.id
: null;
    });
    var dept4=founds.length;
    var founds = $.map(nodestemp, function(val)
{
        return val.group == 'dept5' ? val.id
: null;
    });
    var dept5=founds.length;

    var
pdept1=(dept1/(dept1+dept2+dept3+dept4+dept5))*100;
    var
pdept2=(dept2/(dept1+dept2+dept3+dept4+dept5))*100;
    var
pdept3=(dept3/(dept1+dept2+dept3+dept4+dept5))*100;
    var
pdept4=(dept4/(dept1+dept2+dept3+dept4+dept5))*100;
    var
pdept5=(dept5/(dept1+dept2+dept3+dept4+dept5))*100;

    $('#container').highcharts({

        title: {
            text: ''
        },
        chart: {
            plotBackgroundColor: null,
            plotBorderWidth: null,

```

```

        plotShadow: false
    },
    tooltip: {
        pointFormat: '{series.name}:
<b>{point.percentage:.1f}%</b>'
    },
    exporting: {
        enabled: false
    },
    ...
}

```

Kode Sumber 4. 50 Kode Sumber Proses Menghitung dan Menampilkan Persentase Persebaran Peneliti

4.5.9.4 Proses Pencarian Peneliti

Berikut ini implementasi dari proses pencarian peneliti. Kode Sumber 4. 51 merupakan kode pencarian peneliti.

```

$( "#tags" ).autocomplete({
    source: availableTags,
    select: function(e, ui){

        $('#tags').val(ui.item.value);
        if(spesial) $('#deselecthi').click();
        else
        {
            var found = $.map(nodestemp, function(val) {
                return val.title == $('#tags').val()
? val.id : null;
            });
            if(found[0]==null)
            {
                $.Dialog({
                    'title' :
'Warning!',
                    'content' : 'This
expert not found in the active graph. <br /> Would
you like to find this expert in the special graph?',
                    'position' : {
                        'offsetY' :
'200px'
                    }
                },

```

```

        'buttons' : {
            'Ok' : {
                'action':
function() {

$('#deselecthi').click();

            },
            'Cancel' : {
                'action':
function() {

$('#tags').val('');

            }
        }
    });
}
...
}

```

Kode Sumber 4. 51 Kode Sumber Proses Pencarian Peneliti

4.5.9.5 Proses Menampilkan Peta Kerjasama Peneliti Setiap Area Peneliti

Berikut ini implementasi dari proses menampilkan peta kerjasama peneliti setiap area peneliti. Kode Sumber 4. 52 merupakan kode untuk menampilkan peta kerjasama peneliti setiap area peneliti.

```

click: function () {
    spesial=false;
    var deptst='';
    if(this.name=='Mathematics &
Science') deptst='dept1'; else
    if(this.name=='Industrial
Technology') deptst='dept2'; else
    if(this.name=='Civil Engineering &
Planning') deptst='dept3'; else

```

```

        if(this.name=='Marine Technology')
deptst='dept4'; else
        if(this.name=='Information
Technology') deptst='dept5';
        var found = $.map(nodes,
function(val) {
            return val.group == deptst ? val
: null;
        });
        edgestemp=edges;
        var jsons = new Array();

        var fend= $.map(edgestemp,
function(val) {
            return
(val.fromfaks.toString() == deptst.toString() &&
val.toFaks.toString() == deptst.toString() ) ? val :
null;
        });
        nodestemp=found;
        edgestemp=fend;
        var data = {
            nodes: nodestemp,
            edges: edgestemp
        };
        graph = new vis.Graph(container,
data, options);
        ...
    }

```

**Kode Sumber 4. 52 Proses Menampilkan Peta Kerjasama
Peneliti Setiap Area Peneliti**

4.5.10 Proses Login Manajemen Sistem Repositori Peneliti

Berikut ini implementasi dari proses login manajemen sistem repositori peneliti. Kode Sumber 4. 53 merupakan kode untuk login manajemen sistem repositori peneliti.

```

function index()

```

```
{
    $this->load->library('form_validation');    $this-
>form_validation->set_rules('username', 'Username',
'trim|required|xss_clean');
    $this->form_validation->set_rules('password',
'Password',
'trim|required|xss_clean|callback_check_database');

    if($this->form_validation->run() == FALSE)
    {
$this->load->view('login_view');
    }
    else
    {
        ...}}
}
```

**Kode Sumber 4. 53 Kode Sumber Proses Login Manajemen
Sistem Repositori Peneliti**

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas rangkaian pengujian dan evaluasi terhadap perangkat lunak yang dibangun pada tugas akhir ini. Pengujian dilakukan secara menyeluruh untuk memeriksa apakah fungsi telah berjalan sesuai kebutuhan perangkat lunak. Bab ini mencakup lingkungan, skenario, hasil serta evaluasi terhadap pengujian.

5.1. Lingkungan Pelaksanaan Pengujian

Lingkungan pengujian merupakan perangkat-perangkat yang dilibatkan dalam proses pengujian. Lingkungan pengujian ini menggunakan perangkat komputer yang terhubung dengan jaringan intranet Institut Teknologi Sepuluh Nopember (ITS).

5.2. Dasar Pengujian

Pengujian yang dilakukan berupa pengujian fungsionalitas, pengujian kegunaan perangkat lunak secara langsung oleh pengguna. Pengujian fungsionalitas dilakukan dengan model *black box* untuk masing-masing fungsionalitas dari aplikasi ini. Pengujian ini dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap kebutuhan benar-benar diimplementasikan dan bekerja seperti yang diharapkan.

Pengujian kemudahan perangkat lunak dilakukan dengan pengujian langsung kepada pengguna perangkat lunak. Hasil pengujian diambil dari kuisisioner yang telah disebarakan kepada pengguna perangkat lunak, sebagai titik acuan dari keberhasilan kegunaan perangkat lunak.

5.3. Pengujian Fungsionalitas

Pada subbab ini akan dibahas pengujian perangkat visualisasi peta kerjasama peneliti beserta manajemen visualisasinya untuk

menguji fungsionalitas dari perangkat tersebut. Pengujian didokumentasikan secara sistematis sebagai tolak ukur keberhasilan sistem.

5.3.1 Pengujian Melihat Peta Kerjasama Peneliti Semua Fakultas

Berikut ini pembahasan pengujian melihat peta kerjasama peneliti untuk semua fakultas (area penelitian). Peta kerjasama peneliti untuk semua fakultas adalah tampilan utama pada aplikasi visualisasi peta kerjasama peneliti. Setelah pengguna berhasil membuka aplikasi, maka halaman depan pada graf visualisasi akan menampilkan peta kerjasama peneliti untuk semua fakultas. Skenario rinci pengujian ini dijelaskan pada Tabel 5. 1 dan hasil pengujian ditunjukkan pada Gambar 5. 1.

Tabel 5. 1 Melihat Peta Kerjasama Peneliti Semua Fakultas

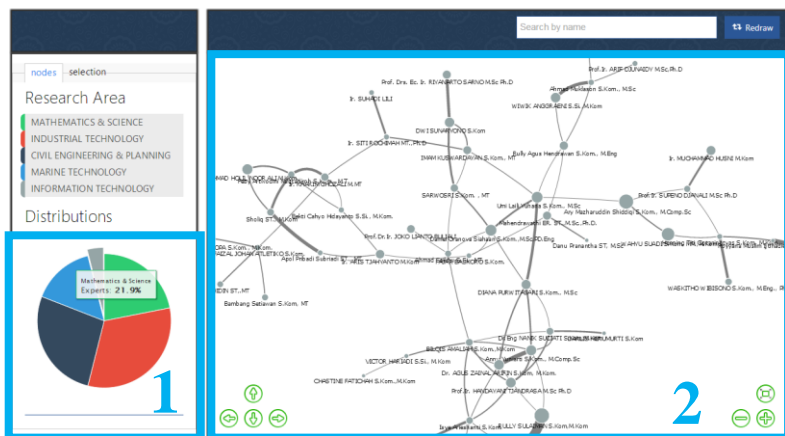
Nomor	PP-01
Nama	Melihat Peta Kerjasama Peneliti Semua Fakultas
Use Case	UC-P01
Tujuan	Memeriksa fungsi untuk peta kerjasama peneliti untuk semua fakultas berfungsi atau tidak.
Kondisi awal	Data peta kerjasama peneliti telah tersimpan.
Skenario	1. Pengguna membuka aplikasi. 2. Pengguna memeriksa pada graf visualisasi apakah peta kerjasama peneliti untuk semua fakultas telah tampil.
Masukan	-

ini dijelaskan pada Tabel 5. 2 dan hasil pengujian ditunjukkan pada Gambar 5.8

Tabel 5. 2 Melihat Peta Kerjasama Peneliti Setiap Fakultas

Nomor	PP-02
Nama	Melihat Peta Kerjasama Peneliti untuk Setiap Fakultas
Use Case	UC-P02
Tujuan	Memeriksa fungsi untuk peta kerjasama peneliti untuk setiap fakultas berfungsi atau tidak.
Kondisi awal	Data peta kerjasama peneliti telah tersimpan.
Skenario	1. Pengguna membuka aplikasi. 2. Pengguna menekan salah satu fakultas pada <i>pie charts</i> .
Masukan	Pilihan fakultas
Keluaran yang diharapkan	Peta kerjasama peneliti untuk fakultas tertentu tampil.
Hasil Pengujian	Berhasil

Pada Gambar 5. 2, kotak nomor 1 menunjukkan bahwa pengguna telah memilih salah satu fakultas yang ingin ditampilkan peta kerjasama peneliti. Sedangkan kotak dengan nomor 2 merupakan hasil dari visualisasi peta kerjasama peneliti dari fakultas yang dipilih.



Gambar 5. 2 Melihat Peta Kerjasama Peneliti Setiap Fakultas

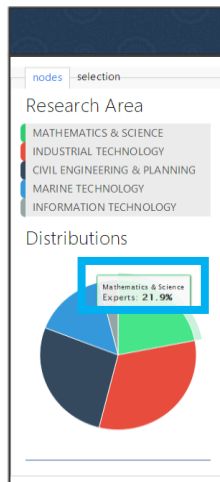
5.3.3 Pengujian Melihat Persentase Persebaran Peneliti Setiap Fakultas

Berikut ini pembahasan pengujian melihat persentase persebaran peneliti setiap fakultas. Persentase persebaran peneliti setiap fakultas ditampilkan ketika pengguna mengarahkan *mouse pointer* ke fakultas tertentu pada *pie charts*. Skenario rinci pengujian ini dijelaskan pada Tabel 5. 3 dan hasil pengujian ditunjukkan pada Gambar 5.8

Tabel 5. 3 Melihat Persebaran Peneliti Setiap Fakultas

Nomor	PP-03
Nama	Melihat Persentase Persebaran Peneliti Setiap Fakultas
Use Case	UC-P03
Tujuan	Memeriksa fungsi untuk menampilkan persentase persebaran peneliti setiap fakultas berfungsi atau tidak.

Kondisi awal	Data peta kerjasama peneliti telah tersimpan.
Skenario	1. Pengguna membuka aplikasi. 2. Pengguna mengarahkan <i>mouse pointer</i> ke salah satu fakultas pada <i>pie charts</i> .
Masukan	Pilihan fakultas
Keluaran yang diharapkan	Persentase persebaran peneliti fakultas yang dipilih tampil.
Hasil Pengujian	Berhasil



Gambar 5. 3 Melihat Persebaran Peneliti Setiap Fakultas

5.3.4 Pengujian Melakukan Pencarian Peneliti

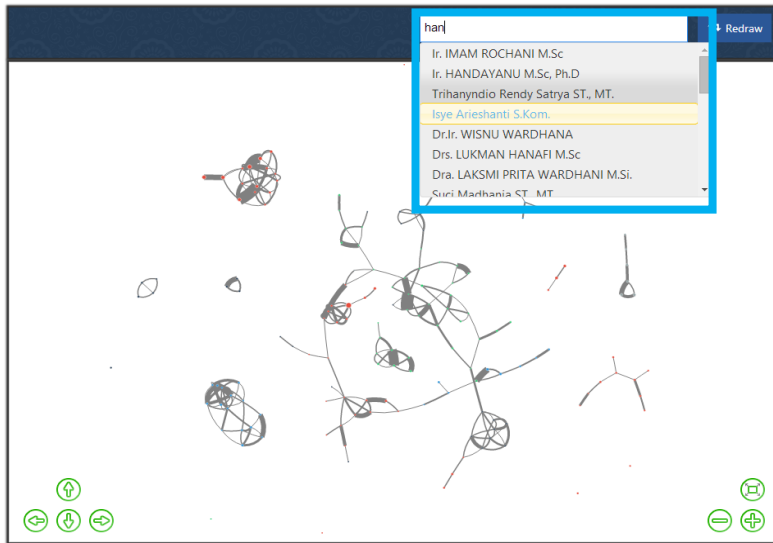
Berikut ini pembahasan pengujian melakukan pencarian peneliti. Pencarian peneliti dilakukan pengguna ketika pengguna ingin menemukan posisi *node* peneliti pada graf visualisasi, atau ingin mengetahui detail informasi peneliti tertentu. Pencarian

dilakukan berdasarkan nama peneliti. Skenario rinci pengujian ini dijelaskan pada Tabel 5. 4 dan hasil pengujian ditunjukkan pada Gambar 5.8.

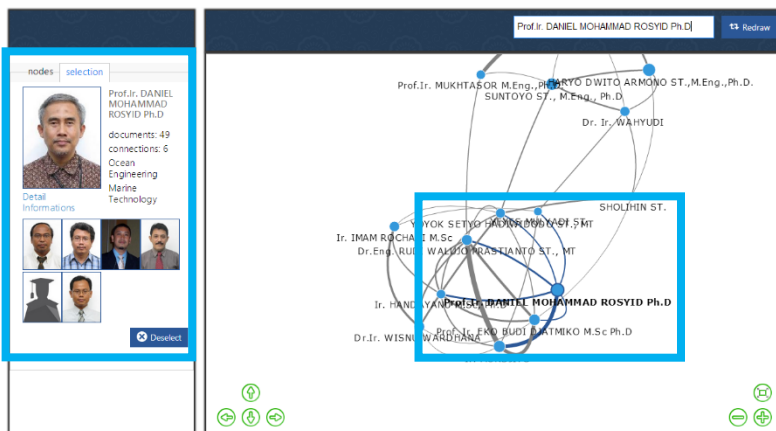
Tabel 5. 4 Melakukan Pencarian Peneliti

Nomor	PP-04
Nama	Melihat Peta Kerjasama Peneliti Semua Fakultas
Use Case	UC-P04
Tujuan	Memeriksa fungsi untuk melakukan pencarian peneliti berfungsi atau tidak.
Kondisi awal	Data peta kerjasama peneliti telah tersimpan.
Skenario	<ol style="list-style-type: none"> 1. Pengguna memasukkan nama peneliti pada form pencarian. 2. Pengguna memilih nama peneliti dari prediksi nama yang muncul.
Masukan	Nama peneliti
Keluaran yang diharapkan	Detail peneliti yang dicari tampil, dan pada graf visualiasi yang aktif, <i>node</i> yang terpilih adalah <i>node</i> peneliti yang dicari.
Hasil Pengujian	Berhasil

Pada saat pengguna memasukkan nama peneliti pada form pencarian, maka muncul prediksi nama-nama sesuai nama peneliti yang dimasukkan, seperti pada Gambar 5. 4. Jika peneliti ditemukan pada visualisasi graf yang sedang aktif maka *node* peneliti pada graf akan terpilih dan detail peneliti akan tampil, seperti pada Gambar 5. 5. Jika pencarian tidak ditemukan pada graf yang sedang aktif, maka akan muncul pilihan untuk menampilkan peneliti pada graf yang lebih spesifik, seperti pada Gambar 5. 6. Hasil pencarian pada graf spesifik, seperti pada Gambar 5. 7.



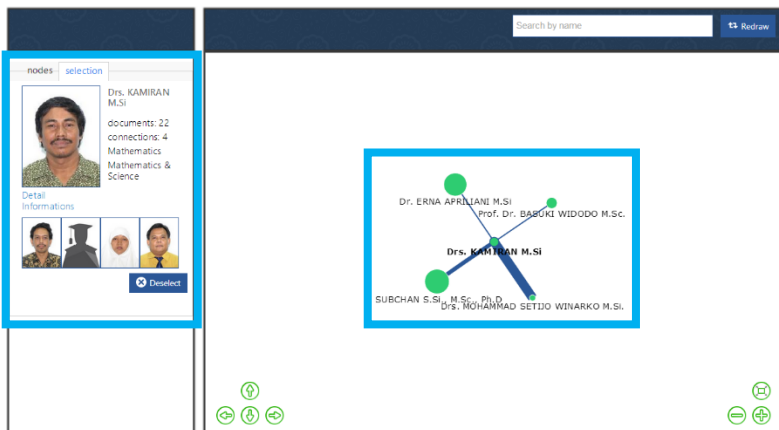
Gambar 5. 4 Memasukkan Nama Peneliti



Gambar 5. 5 Pencarian Peneliti Ditemukan pada Graf Aktif



Gambar 5. 6 Pencarian Tidak Ditemukan pada Graf Aktif



Gambar 5. 7 Pencarian Peneliti Ditemukan pada Graf Spesifik

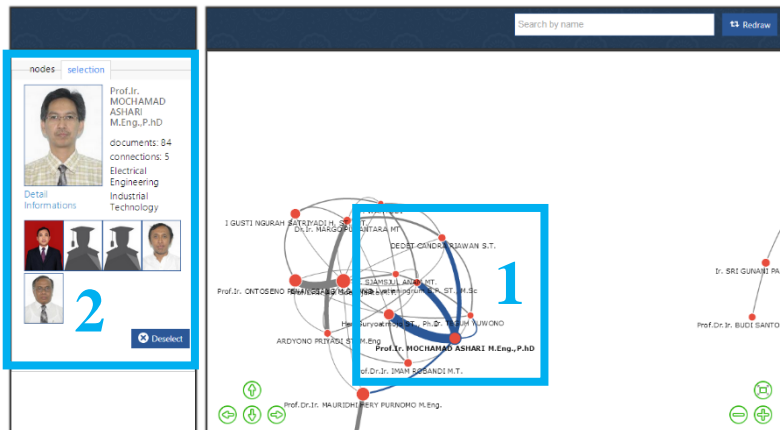
5.3.5 Pengujian Melihat Detail Peneliti

Berikut ini pembahasan pengujian melihat detail peneliti. Detail peneliti akan ditampilkan ketika salah satu *node* pada graf visualisasi dipilih. Skenario rinci pengujian ini dijelaskan pada Tabel 5. 5 dan hasil pengujian ditunjukkan pada Gambar 5.8

Tabel 5. 5 Melihat Detail Peneliti

Nomor	PP-05
Nama	Melihat Detail Peneliti
Use Case	UC-P05
Tujuan	Memeriksa fungsi untuk menampilkan detail peneliti berfungsi atau tidak.
Kondisi awal	Data peta kerjasama peneliti telah tersimpan.
Skenario	Pengguna memilih salah satu <i>node</i> pada graf visualisasi.
Masukan	Pilihan <i>node</i> pada graf visulisasi
Keluaran yang diharapkan	Detail peneliti tampil.
Hasil Pengujian	Berhasil

Pada Gambar 5. 8, setelah pengguna memilih salah satu *node* pada graf visualisasi, seperti pada kotak nomor 1, maka akan tampil detail peneliti seperti yang ditunjukkan pada kotak nomor 2.



Gambar 5. 8 Melihat Detail Peneliti

5.3.6 Pengujian Memperbarui Data Peta Kerjasama Peneliti

Berikut ini pembahasan pengujian memperbarui data peta kerjasama peneliti. Pembaruan data peta kerjasama peneliti ini dilakukan oleh administrator. Pembaruan dilakukan ketika terdapat data penelitian baru atau ada peneliti baru, sehingga perhitungan peta kerjasama peneliti harus diperbarui. Skenario rinci pengujian ini dijelaskan pada Tabel 5. 6 dan hasil pengujian ditunjukkan pada Gambar 5.8

Tabel 5. 6 Memperbarui Data Peta Kerjasama Peneliti

Nomor	PP-06
Nama	Memperbarui Data Peta Kerjasama Peneliti
Use Case	UC-P06
Tujuan	Memeriksa fungsi untuk memperbarui data peta kerjasama peneliti berfungsi atau tidak.

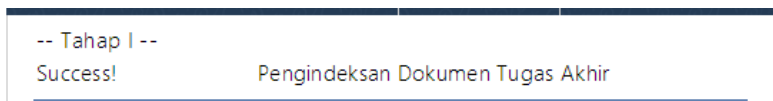
Kondisi awal	Data peta kerjasama peneliti masih belum diperbarui
Skenario	<ol style="list-style-type: none"> 1. Administrator masuk ke manajemen Sistem Repositori Peneliti. 2. Administrator memilih bagian manajemen visualisasi. 3. Administrator melakukan tahap-tahap pembaruan data peta kerjasama peneliti.
Masukan	-
Keluaran yang diharapkan	Data peta kerjasama peneliti pada <i>file</i> JSON diperbarui.
Hasil Pengujian	Berhasil

Gambar 5. 9 menunjukkan tahapan-tahapan untuk memperbarui data peta kerjasama peneliti. Terdapat 7 tahapan, yaitu pengindeksan dokumen tugas akhir, menghilangkan *stopwords* dan melakukan *stemming*, pembobotan kata, penghitungan bobot relasi topik antar peneliti, penghitungan bobot kerjasama penelitian antar peneliti, penghitungan bobot kerjasama berdasarkan topik penelitian, dan *generate* pemodelan graf peta kerjasama peneliti dalam bentuk *file* JSON.

Tahap 1 seperti pada Gambar 5. 10, tahap 2 seperti pada Gambar 5. 11, tahap 3 seperti pada Gambar 5. 12, tahap 4 seperti pada Gambar 5. 13, tahap 5 seperti pada Gambar 5. 14, tahap 6 seperti pada Gambar 5. 15, dan tahap 3 seperti pada Gambar 5. 16.



Gambar 5. 9 Tahapan Pembaruan Data Kerjasama Peneliti Pada Manajemen Visualisasi



Gambar 5. 10 Manajemen Visualisasi Tahap 1

-- Tahap II --	
Success!	Menghilangkan stopwords, Melakukan Stemming

Gambar 5. 11 Manajemen Visualisasi Tahap 2

-- Tahap III --	
Success!	Pembobotan Kata

Gambar 5. 12 Manajemen Visualisasi Tahap 3

-- Tahap IV --	
Success!	Penghitungan Bobot Relasi Topik Antar Peneliti

Gambar 5. 13 Manajemen Visualisasi Tahap 4

-- Tahap V --	
Success!	Penghitungan Bobot Kerjasama Penelitian antar Peneliti

Gambar 5. 14 Manajemen Visualisasi Tahap 5

-- Tahap VI --	
Success!	Penghitungan Bobot Kerjasama Peneliti Berdasarkan Topik Penelitian

Gambar 5. 15 Manajemen Visualisasi Tahap 6

-- Tahap VII --	
Success!	Generate Pemodelan Peta Kerjasama Peneliti dalam bentuk File JSON

Gambar 5. 16 Manajemen Visualisasi Tahap 7

5.4 Pengujian Kegunaan

Pengujian kegunaan dilakukan dengan melibatkan beberapa pengguna dari aplikasi ini. Para pengguna yang terlibat dalam pengujian ini selanjutnya disebut sebagai responden. Para responden diberikan keleluasaan untuk menjalankan aplikasi pada perangkat komputer.

5.4.1 Kriteria Responden

Masing-masing responden menjalankan perangkat lunak pada perangkat komputer. Perangkat komputer yang digunakan oleh responden terhubung dengan jaringan intranet Institut Teknologi Sepuluh Nopember (ITS). Adapun responden yang terlibat dalam pengujian dari aplikasi ini berasal dari 3 kriteria, yaitu : dosen, mahasiswa, dan masyarakat umum.

5.4.2 Lingkungan Pengujian Kegunaan

Masing-masing responden menjalankan perangkat lunak pada perangkat komputer yang terhubung pada jaringan intranet ITS. Perangkat komputer yang digunakan sudah tersedia *web browser* untuk mengakses perangkat lunak.

5.4.3 Skenario Pengujian Kegunaan

Dalam melakukan pengujian kegunaan aplikasi, masing-masing responden diminta melakukan eksplorasi terhadap kaskas visualisasi peta kerjasama peneliti seperti melihat peneliti-peneliti yang mirip topik penelitiannya, melakukan pencarian peneliti, dll.

Dalam memberikan penilaian terhadap aplikasi, responden mengisi formulir penilaian yang telah disediakan untuk pengujian ini. Formulir penilaian pengujian kegunaan aplikasi yang digunakan dapat dilihat pada Lampiran B tentang Kuisioner Pengujian Kegunaan.

5.4.4 Daftar Responden

Terdapat 17 orang yang menjadi responden dalam pengujian kegunaan visualisasi peta kerjasama peneliti. Responden berasal dari dosen/peneliti, mahasiswa, dan masyarakat umum, seperti yang ditunjukkan pada Tabel 5. 7.

Tabel 5. 7 Responden Pengujian Kegunaan Visualisasi Peta Kerjasama Peneliti

No.	Nama	Status	Lingkungan Pengujian (Browser/Sistem Operasi)	Tanggal Pengujian
1	Fadhilah Kurnia Putri	Mahasiswa S1	Google Chrome v 31.0 / Windows 8	1 Juli 2014
2	Gandha Hafesha	Mahasiswa S1	Google Chrome v 35.0 / Windows 8.1	6 Juli 2014
3	Anita Susanti	Mahasiswa S1	Mozilla Firefox v 30.0 / Windows 7	29 Juni 2014
4	Luluk Eko Mawati	Mahasiswa S1	Google Chrome v 29.0 / Windows 7	29 Juni 2014
5	Hardika Khusnuliawati	Mahasiswa S1	Mozilla Firefox v 10.0 / Windows 7	29 Juni 2014
6	Fahmi Hidayatullah	Mahasiswa S1	Google Chrome v 31.0 / Windows 8	29 Juni 2014
7	Ika Astutik	Mahasiswa S1	Mozilla Firefox v 30.0 / Windows 7	29 Juni 2014
8	Fajar Hidayat	Mahasiswa S1	Mozilla Firefox v 29.0 / Windows 8	29 Juni 2014
9	Yolanda Septiana Dewi	Mahasiswa S1	Google Chrome v 35.0 / Windows 7	29 Juni 2014
10	Fadlika Dita Nurjanto	Mahasiswa S1	Google Chrome v 35.0 / Windows 8	29 Juni 2014
11	Qotrun Nada Haroen	Mahasiswa S1	Google Chrome v 34.0 / Windows 7	29 Juni 2014
12	M. Aminudin Rahman	Mahasiswa S1	Mozilla Firefox v 29.0 / Windows 8	30 Juni 2014

13	Muamar Agus Salim	Mahasiswa S1	Mozilla Firefox v 30.0 / Windows 7	29 Juni 2014
14	Muhammad Irfan	Mahasiswa S2	Mozilla Firefox v 30.0 / Windows 7	1 Juli 2014
15	Dosen Swanjaya	Dosen	Mozilla Firefox v 29.0 / Windows Server 2003	1 Juli 2014
16	Agro Wisudawan	Dosen	Mozilla Firefox v 30.0 / Windows 7	30 Juni 2014
17	Sukma Arbianto S	Umum	Google Chrome v 35.0 / Windows 8	29 Juni 2014

5.4.5 Hasil Pengujian Kegunaan

Pada penilaian pengujian kegunaan, terdapat rentang angka antara 1 sampai dengan 10 untuk masing-masing indikator penilaian pertanyaan, dimana angka 1 merepresentasikan sangat tidak baik dan angka 10 merepresentasikan sangat baik.

5.3.1.1 Rekapitulasi Penilaian Antarmuka Pengguna

Antarmuka pengguna merupakan salah satu aspek yang dinilai pada pengujian kegunaan aplikasi ini. Rekapitulasi penilaian pengujian terhadap antarmuka pengguna terdapat pada Tabel 5. 8.

Tabel 5. 8 Rekapitulasi Penilaian Antarmuka Pengguna

No	Penilaian										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
1	0	0	0	0	0	0	3	8	6	0	8.17647059
2	0	0	0	0	0	3	3	9	2	0	7.58823529
3	0	0	0	0	0	2	2	8	5	0	7.94117647
Nilai Akhir											7.90196078

Tabel 5. 8 di atas menunjukkan bahwa antarmuka pengguna dari perangkat lunak visualisasi peta kerjasama peneliti memiliki nilai rata-rata 7,90196078.

5.3.1.2 Rekapitulasi Penilaian Informasi Peta Kerjasama Peneliti

Informasi yang disajikan pada graf visualisasi peta kerjasama peneliti merupakan salah satu aspek yang dinilai pada pengujian kegunaan aplikasi ini. Rekapitulasi penilaian pengujian terhadap informasi graf peta kerjasama peneliti terdapat pada Tabel 5. 9.

Tabel 5. 9 Rekapitulasi Penilaian Informasi Graf Peta Kerjasama Peneliti

No	Penilaian										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
1	0	0	0	0	0	0	3	5	7	2	8.470588
2	0	0	0	0	0	0	3	9	4	1	8.176471
3	0	0	0	0	0	0	7	9	0	0	7.117647
Nilai Akhir											7.921569

Tabel 5. 9 di atas menunjukkan bahwa informasi graf peta kerjasama peneliti memiliki nilai rata-rata 7,921569.

5.3.1.3 Rekapitulasi Penilaian Informasi Peneliti

Ketepatan informasi detail peneliti merupakan salah satu aspek yang dinilai pada pengujian kegunaan aplikasi ini. Rekapitulasi penilaian pengujian terhadap ketepatan informasi detail peneliti terdapat pada Tabel 5. 10.

Tabel 5. 10 Rekapitulasi Penilaian Informasi Ketepatan Lokasi

No	Penilaian										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
1	0	0	0	0	0	0	3	9	4	1	8.17647059
2	0	0	0	0	0	0	3	8	6	0	8.17647059
Nilai Akhir											8.17647059

Tabel 5. 10 di atas menunjukkan bahwa ketepatan informasi detail peneliti pada kaskas visualisasi peta kerjasama peneliti memiliki nilai rata-rata 8,17647059.

5.4 Evaluasi Pengujian

Berdasarkan hasil pengujian fungsionalitas dan pengujian kegunaan, maka dapat dilakukan evaluasi sebagai berikut.

5.4.1 Evaluasi Pengujian Fungsionalitas

Pengujian fungsionalitas yang telah dilakukan memberikan hasil yang sesuai dengan skenario yang telah direncanakan. Evaluasi pengujian pada masing-masing fungsionalitas dijelaskan dengan sebagai berikut.

1. Pengujian melihat peta kerjasama peneliti untuk semua fakultas sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-01 yang memberikan informasi bahwa proses melihat peta kerjasama peneliti untuk semua fakultas berjalan dengan benar.
2. Pengujian melihat peta kerjasama peneliti untuk setiap fakultas sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-02 yang memberikan informasi bahwa proses melihat peta kerjasama peneliti untuk setiap fakultas berjalan dengan benar.

3. Pengujian melihat persentase persebaran peneliti untuk setiap fakultas sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-03 yang memberikan informasi bahwa proses melihat persentase persebaran peneliti untuk setiap fakultas berjalan dengan benar.
4. Pengujian melakukan pencarian peneliti sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-04 yang memberikan informasi bahwa proses melakukan pencarian peneliti berjalan dengan benar.
5. Pengujian melihat detail peneliti sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-05 yang memberikan informasi bahwa proses melihat detail peneliti berjalan dengan benar.
6. Pengujian memperbarui data peta kerjasama peneliti sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PP-06 yang memberikan informasi bahwa proses memperbarui data peta kerjasama peneliti berjalan dengan benar.

5.4.2 Evaluasi Pengujian Kegunaan

Untuk mengevaluasi pengujian kegunaan perangkat lunak, maka dilakukan rekapitulasi akhir. Rekapitulasi akhir menghasilkan nilai presentase terhadap aspek-aspek yang dinilai pada pengujian kegunaan. Rekapitulasi akhir pada dilihat pada Tabel 5. 11.

Tabel 5. 11 Rekapitulasi Akhir Pengujian Kegunaan

No.	Aspek Pengujian	Nilai Akhir	Persentase
1.	Antarmuka Pengguna	7,90196078	79,0196078%
2.	Informasi Graf Peta Kerjasama Peneliti	7,921569	79,21569%
3.	Informasi Peneliti	8,17647059	81,7647059%

Dari Tabel 5. 11 dapat dilihat bahwa aspek-aspek yang dinilai menghasilkan nilai-nilai presentase. Penilaian terhadap antarmuka menghasilkan persentase 79,0196078 %, penilaian terhadap informasi graf peta kerjasama peneliti menghasilkan persentase 79,21569 %, dan penilaian terhadap informasi informasi peneliti menghasilkan persentase 81,7647059 %. Ketiga aspek menunjukkan persentase nilai yang lebih dari 75% sehingga dapat disimpulkan bahwa dari segi kegunaan, perangkat lunak yang dibangun dalam Tugas Akhir ini memberikan hasil yang cukup baik.

BAB VI

PENUTUP

Pada bab ini akan dijelaskan kesimpulan yang dapat diambil dalam pengerjaan tugas akhir dan saran tentang kemungkinan pengembangan yang dapat dilakukan pada tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari proses analisa, perancangan, implementasi dan pengujian perangkat lunak visualisasi peta kerjasama peneliti dapat ditarik kesimpulan sebagai berikut.

1. Aplikasi visualisasi peta kerjasama peneliti yang dibangun dapat menampilkan peta kerjasama peneliti berdasarkan topik penelitian menggunakan pemodelan graf berbasis web. Data penelitian yang diolah berasal dari 11.824 dokumen penelitian tugas akhir dan 746 peneliti, dengan batas *threshold* hubungan kerjasama berdasarkan kemiripan topik sebesar 0,009.
2. Aplikasi dapat menampilkan data peta kerjasama peneliti yang dapat berubah-ubah sesuai dengan data penelitian terbaru.
3. Aplikasi dapat menampilkan peneliti-peneliti berdasarkan kemiripan topik penelitiannya.

6.2. Saran

Berikut ini beberapa saran untuk kemungkinan pengembangan perangkat lunak visualisasi peta kerjasama peneliti di masa mendatang berdasarkan hasil rancangan, implementasi dan uji coba yang telah dilakukan.

1. Pada visualisasi peta kerjasama peneliti saat ini masih terbatas pada penggunaan data penelitian yang berasal dari tugas akhir. Sehingga perlu dikembangkan pula untuk jenis dokumen penelitian yang lain.
2. Pada aplikasi ini hubungan kerjasama antar peneliti masih bersifat *one-to-one*, artinya setiap peneliti pasti memiliki hubungan langsung dengan peneliti lain. Sehingga ini mengakibatkan visualisasi yang ditampilkan tidak informatif karena banyaknya hubungan (garis) antar peneliti. Untuk itu perlu solusi yang tepat agar visualisasi lebih informatif.
3. Pada aplikasi ini, hubungan kerjasama berdasarkan kemiripan topik antar peneliti belum dilakukan uji kebenaran untuk mengetahui validitasnya dalam kondisi yang sebenarnya. Hal ini dikarenakan belum ditemukan metode yang tepat untuk melakukan uji kebenaran tersebut. Sehingga diperlukan sebuah solusi yang tepat untuk menguji kebenaran data hubungan peta kerjasama peneliti yang ditampilkan pada aplikasi ini.

DAFTAR PUSTAKA

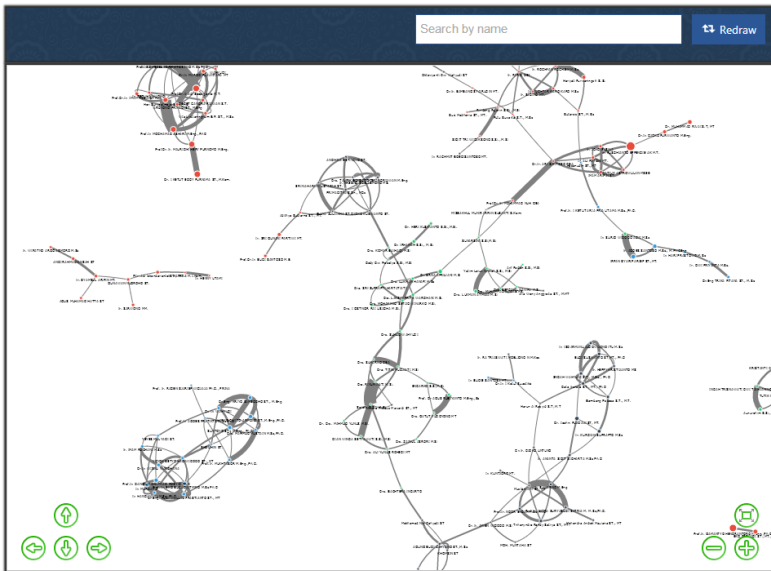
- [1] H. Gunawan, "Hendra Gunawan FMIPA ITB Personal Blog," [Online]. Available: <http://personal.fmipa.itb.ac.id/hgunawan/files/2010/12/peta-kekuatan-penelitian-dan-deposito-pengetahuan-indonesia-10-des-2010.pdf>. [Accessed 20 June 2014].
- [2] C. Belter, "Information Today, Inc. ITI Blogs," NOAA Central Library, [Online]. Available: <http://www.infotoday.com/online/may12/Belter-Visualizing-Networks-of-Scientific-Research.shtml>. [Accessed 20 June 2014].
- [3] Institut Teknologi Sepuluh Nopember, "Personal Dosen ITS," Institut Teknologi Sepuluh Nopember, [Online]. Available: <http://personal.its.ac.id/>. [Accessed 25 June 2014].
- [4] EllisLab, Inc., "What is CodeIgniter?," EllisLab, Inc., 2012. [Online]. Available: <http://ellislab.com/codeigniter>. [Accessed 24 June 2014].
- [5] Zend Technologies Ltd. , "Query Language - Zend_Search_Lucene - Zend Framework," Zend Technologies Ltd. , 2005. [Online]. Available: <http://framework.zend.com/manual/1.10/en/zend.search.lucene.query-language.html>. [Accessed 3 March 2014].
- [6] A. B.V, "vis.js is a dynamic, browser-based visualization library," 2010. [Online]. Available: <http://visjs.org/>. [Accessed 24 June 2014].
- [7] A. Alamsyah, "IlmuKomputer.com," [Online]. Available: <http://ikc.dinus.ac.id/umum/andry/andry-javascript.pdf>. [Accessed 24 June 2014].
- [8] json.org, "Introducing JSON," json.org, [Online]. Available: <http://json.org/>. [Accessed 24 June 2014].

- [9] T. Rollek'e, T. Tsikrika and G. Kazai, "A General Matrix Framework for Modelling Information Retrieval," [Online]. Available: http://cir.dcs.uni-pannon.hu/cikkek/rolleke_paper.pdf. [Accessed 25 June 2014].
- [10] A. Brown, "Anders Breivik's spider web of hate," Guardian News and Media Limited , 7 September 2011. [Online]. Available: <http://www.theguardian.com/commentisfree/2011/sep/07/anders-breivik-hate-manifesto>. [Accessed 5 March 2014].
- [11] C. A. R. Pinheiro, Social Network Analysis in Telecommunications, John Wiley & Sons, 2011.
- [12] A. Abraham, A.-E. Hassanien and V. Snášel, Computational Social Network Analysis: Trends, Tools and Research Advances, 2009.
- [13] Orgnet, LLC, "A Brief Introduction to Social Network Analysis by Orgnet, LLC," Orgnet, LLC, [Online]. Available: <http://www.orgnet.com/sna.html>. [Accessed 24 June 2014].
- [14] H. C. WU, R. W. P. LUK, K. F. WONG and K. L. KWOK, "Interpreting TF-IDF Term Weights as Making Relevance Decisions," in *ACM Transactions on Information Systems*, 2008.

LAMPIRAN A – FORMULIR PENENTUAN *THRESHOLD* PETA KERJASAMA PENELITI

Rancang Bangun Visualisasi Peta Kerjasama Peneliti Berdasarkan Topik Penelitian pada Sistem Repositori Peneliti menggunakan Pemodelan Graf

1. Dari visualisasi peta kerjasama di bawah ini, manakah yang menurut Anda paling mudah dipahami? (Berdasarkan jumlah *node* yang tampil pada graf visualisasi)



Gambar A. 1 Graf visualisasi peta kerjasama peneliti dengan 178 node

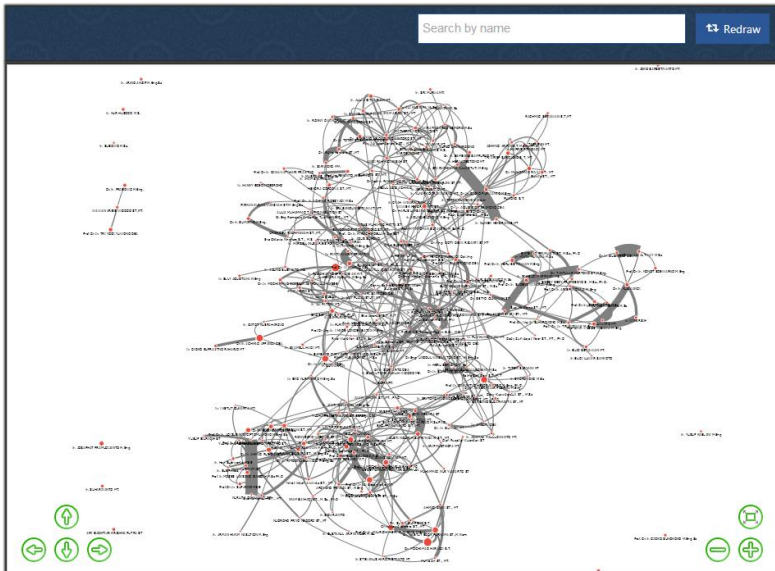


Gambar A. 2 Graf visualisasi peta kerjasama peneliti dengan 400 node

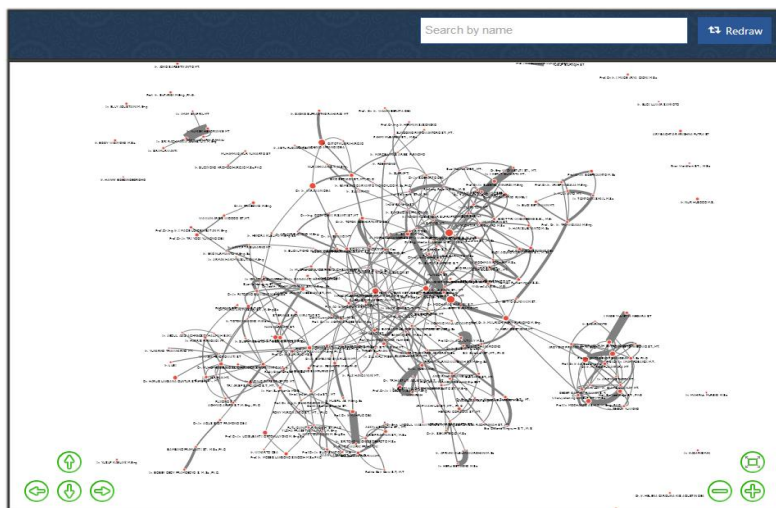


Gambar A. 3 Graf visualisasi peta kerjasama peneliti dengan 700 node

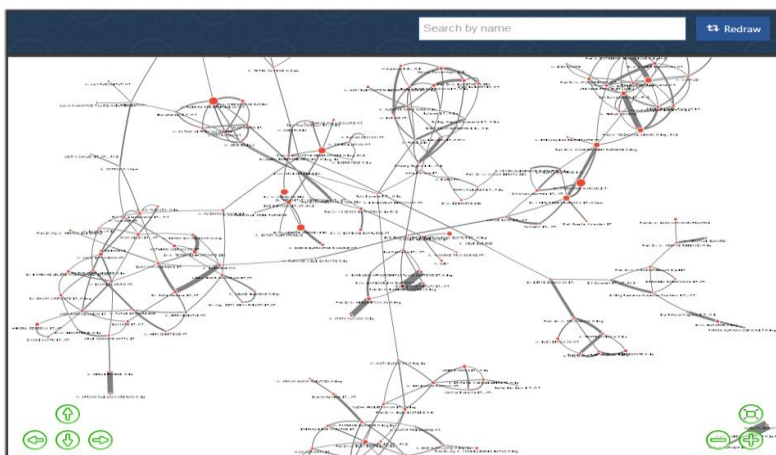
2. Dari visualisasi peta kerjasama di bawah ini, manakah yang menurut Anda paling mudah dipahami? (Berdasarkan *threshold* (batas) bobot hubungan antar peneliti pada peta kerjasama peneliti Fakultas Teknologi Industri)



Gambar A. 4 Peta kerjasama peneliti dengan *threshold* bobot hubungan antar peneliti 0.005



Gambar A. 5 Peta kerjasama peneliti dengan *threshold* bobot hubungan antar peneliti 0.007



Gambar A. 6 Peta kerjasama peneliti dengan *threshold* bobot hubungan antar peneliti 0.009

LAMPIRAN B – FORMULIR PENILAIAN PENGUJIAN KEGUNAAN

Rancang Bangun Visualisasi Peta Kerjasama Peneliti Berdasarkan Topik Penelitian pada Sistem Repositori Peneliti menggunakan Pemodelan Graf

Data Responden

Tanggal :
Nama :
Status/Pekerjaan :

Lingkungan Pengujian

Tipe Browser :
Versi Browser :
Sistem Operasi :

A. Antarmuka Pengguna

1. Apakah aplikasi ini memiliki antarmuka pengguna yang menarik?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

2. Apakah antarmuka aplikasi ini mudah dipahami?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

3. Apakah aplikasi ini sudah memiliki tata letak yang baik?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

B. Informasi Graf Peta Kerjasama Peneliti

1. Apakah aplikasi ini membantu Anda mengetahui peta kerjasama peneliti di ITS?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

2. Apakah aplikasi ini menyajikan graf peta kerjasama peneliti secara informatif?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

3. Apakah peluang kerjasama penelitian dapat diprediksi melalui graf peta kerjasama peneliti pada aplikasi ini?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

C. Informasi Peneliti

1. Apakah aplikasi ini sudah menampilkan detail informasi peneliti sesuai dengan data peneliti yang Anda ketahui?

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

2. Apakah aplikasi ini sudah dapat membantu Anda menemukan peneliti yang mirip topik penelitiannya?

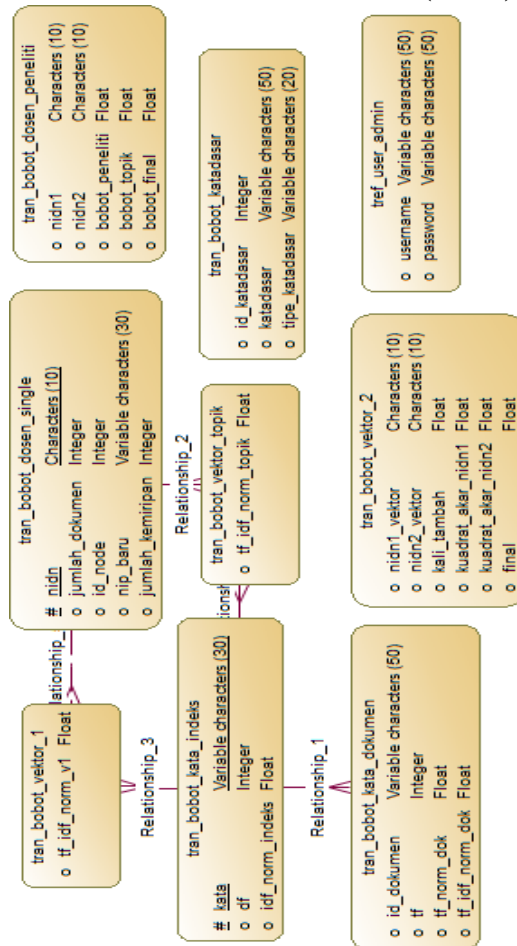
1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

D. Saran dan Kritik

1. Saran Anda untuk aplikasi ini,

2. Kritik Anda untuk aplikasi ini,

LAMPIRAN C – PERANCANGAN BASIS DATA CONCEPTUAL DATA MODEL (CDM) DAN PHYSICAL DATA MODEL (PDM)



**Gambar C. 1 Conceptual Data Model (CDM) Aplikasi
Visualisasi Peta Kerjasama Peneliti**

LAMPIRAN D – HASIL SURVEI PENENTUAN THRESHOLD PETA KERJASAMA PENELITIAN

Berdasarkan hasil survei ke pengguna tentang penentuan *threshold* peta kerjasama peneliti seperti pada Lampiran A, maka hasil survei tersebut adalah seperti yang ditunjukkan pada Tabel D. 1 dan Tabel D. 2.

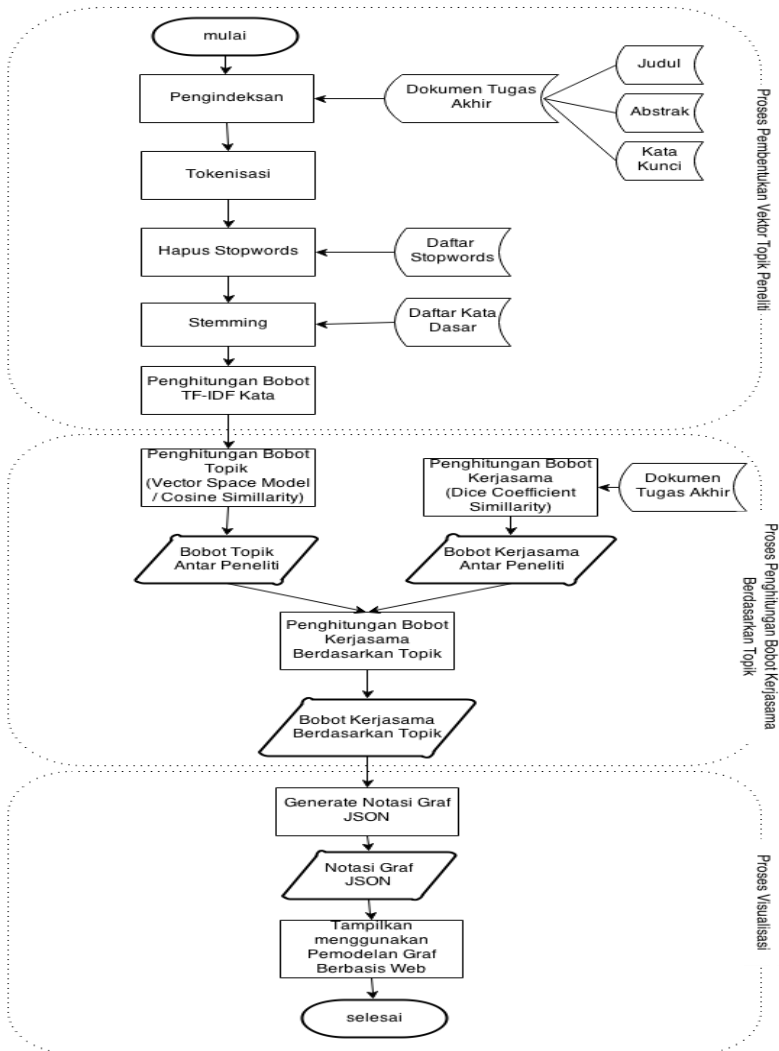
Tabel D. 1 Hasil Survei Penentuan *Threshold* Bobot Antar Peneliti

	Threshold Bobot			Persentase Pilihan		
	0.005	0.007	0.009	0.005	0.007	0.009
Jumlah Responden	0	0	9	0 %	0 %	100 %
Total Responden	9					

Tabel D. 2 Hasil Survei Penentuan Jumlah Mode Pada Graf Visualisasi

	Jumlah Node			Persentase Pilihan		
	178	400	700	178	400	700
Jumlah Responden	9	0	0	100 %	0 %	0 %
Total Responden	9					

LAMPIRAN E – DIAGRAM ALIR KAKAS VISUALISASI



Gambar E. 1 Diagram Alir Kakas Visualisasi

BIODATA PENULIS



Penulis lahir di Sarolangun, 23 Agustus 1991. Penulis menempuh pendidikan di SD Bangun Jaya (Sarolangun, Jambi), SDN 02 Jatiroto (Pati), SMPN 1 Kayen (Pati) dan SMAN 1 Pati. Penulis melanjutkan pendidikan sarjana di Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya. Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Rekayasa Perangkat Lunak (*Software Engineering*). Selama kuliah penulis aktif dalam berbagai kegiatan dan organisasi baik itu akademik maupun nonakademik.

Pada dunia akademik, penulis aktif sebagai asisten praktikum mata kuliah Pemrograman Terstruktur, Algoritma dan Struktur Data. Selain itu penulis juga aktif sebagai Administrator Laboratorium Sistem Cerdas. Pada dunia nonakademik, penulis diberi amanah sebagai Ketua Departemen Kaderisasi Keluarga Muslim Informatika (KMI) pada periode kepengurusan 2012-2013.

Penulis tertarik pada pengembangan permainan, *augmented reality*, aplikasi *mobile*, dan pengembangan *web*. Penulis dapat dihubungi melalui alamat email agus.nugroho@live.com.